DOCUMENT RESUME

ED 163 997 IR 006 792

AUTHOR	Hughes, John; And Others
TITLE	Functional Specifications for Computer Aided Training Systems Development and Management (CATSDM) Support Functions. Final Report.
INSTITUTION	Courseware, Inc., San Diego, Calif.
SPONS AGENCY	Naval Training Equipment Center, Orlando, Fla.
REPORT NO	NAVTRAEQUIPCEN-77-C-0018-1
PUB DATE	Jul 78
NOTE	123p.
EDRS PRICE	MF-$0.83 HC-$6.01 Plus Postage.
DESCRIPTORS	*Computer Assisted Instruction; Computer Managed Instruction; Educational Research; *Educational Specifications; Instructional Design; *Instructional Systems; *Management Information Systems; Military Organizations; Models; Program Descriptions; Program Development; *Training Techniques
IDENTIFIERS	*Instructional Development

ABSTRACT

This report provides a description of a Computer Aided Training System Development and Management (CATSDM) environment based on state-of-the-art hardware and software technology, and including recommendations for off the shelf systems to be utilized as a starting point in addressing the particular systematic training and instruction design and development model described in MIL STD 961 (often referred to as the Fleet Aviation Instructional Systems Design Model). The functional specifications describe a system of computer programs to assist in successfully and reliably carrying out the Instructional System Development (ISD) process. These programs are designed to lend structure to the activities of persons involved in ISD, as well as to provide a wide range of support functions, including needed data collection and manipulation capabilities for all phases of the process. The remainder of the document deals in detail with the components of CATSDM, conception and development of these components, and recommendations and conclusions made in light of the analysis of existing systems. (Author/CMV)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED163997

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

FUNCTIONAL SPECIFICATIONS FOR COMPUTER AIDED TRAINING SYSTEMS DEVELOPMENT AND MANAGEMENT (CATSDM) SUPPORT FUNCTIONS

BY

John Hughes, Fred O'Neal, Greg Kearsley, Harriet O'Neal
Mike Fee, Linda Rodi and Jan Hermanns

Courseware, Inc.
10075 Carroll Canyon Road
San Diego, California 92131

Final Report, January 1977- May 1978

July 1978

Technical Report NAVTRAEQUIPCEN 77-C-0018-1

Naval Training Equipment Center
Orlando, Florida 32813

IR006792

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
NAVTRAEQUIPCEN 77-C-0018-1

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Functional Specifications for Computer Aided Training Systems Development and Management (CATSDM) Support Functions

5. TYPE OF REPORT & PERIOD COVERED
January 1977 - July 1978

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
John Hughes, Fred O'Neal, Greg Kearsley
Harriet O'Neal, Mike Fee, Linda Rodi,
Jan Hermanns

8. CONTRACT OR GRANT NUMBER(s)
N61339-77-C-0018

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Courseware, Inc.
10075 Carroll Canyon Road
San Diego, CA 92131

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
6754-1P3TQ

11. CONTROLLING OFFICE NAME AND ADDRESS
Naval Training Equipment Center
Code N-71
Orlando, FL 32813

12. REPORT DATE
July 1978

13. NUMBER OF PAGES
121

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ISD, Training Development, Instructional Development, Management, Training Design, Instructional Design, Computer Based Management Systems

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The scope and complexity of major instructional design and development programs and the sophistication and detailed level of definition of particular training development models utilized have greatly increased. The advanced planning and day to day management necessary for such programs have become commensurately more difficult and complex. The interactions between the many resource personnel and schedule requirements involvedincrease the difficulty of identifying specific sources of problems and responding to them without causing problems elsewhere. In addition, the models being used tend to have an increasing number of engineering characteristics rather than artistic attributes. That is, they are characterized by increasingly well defined procedures and techniques which offer the potential for much better quality control consistency of output across different personnel, and they simplify the problem of training and standardizing approaches for development staffs. In response to the challenges of these emerging models interest is greatly increasing in comprehensive real-time integrated information management systems which incorporate a variety of flexible management projection and simulation capabilities, and which include rich arrays of standardized protocols, staff training materials, and interactive job aids. This report is a functional specification for a Computer Aided Training System Development and Management (CATSDM) environment based on state-of-the-art hardware and software technology, and including recommendations for off the shelf systems to utilize as a starting point in addressing the particular systematic training and instruction design and development model described in MIL STD 961 (often referred to as the Fleet Aviation ISD Model).

TABLE OF CONTENTS

Section		Page
I	INTRODUCTION	5
	Purpose of this Document	5
	Overview of the Contents	5
	Description of Methods	7
II	DESCRIPTION OF PROGRAMS	9
	Overview of the System of Programs	9
	Analysis Phase Programs and Data Bases	9
	Design Phase Programs and Data Bases	10
	Development Phase Programs and Data Bases	11
	Implementation Phase Programs and Data Bases	12
	Programs Peripheral to Mainstream ISD Activities	12
	Description of Program Capabilities	13
	Definitions of Standard Capabilities	13
	Description of Functions of Each Program	24
	Analysis Phase Programs	24
	Design Phase Programs	29
	Development Phase Programs	35
	Implementation Phase Programs	38
	Programs Peripheral to Mainstream ISD Activities	40
	Description of the Data Bases	46
	Analysis Phase Data Bases	47
	Design Phase Data Bases	48
	Development Phase Data Bases	48
	Implementation Phase Data Bases	48
	Data Bases Peripheral to Mainstream ISD Activities	49
III	ANALYSIS OF VTS IN LIGHT OF CATSDM RECOMMENDATIONS	51
	Overview	51
	Comparison of Standard Program Capabilities	51
	Description of Program Capabilities	51
	Comparison of Functions for Each Program	54
	Course Authoring Functions	54
	Functions Which Create Reports	54
	Student Entry Level Program	55

Section Page
Existing Materials Evaluation Program 56
Historical Trainer Data Program 56
Progress Monitoring Program 56
Computer-Managed Instruction Program 57
Computer-Assisted Instruction Program 57
Quality Control Plan 58
Comparison of Data Base Contents and Structure 58
Description of the Data Base 58
Student Entry Level Data Base 58
Task Listing Data Base 59
Objectives Hierarchy Data Base 59
Course Syllabi Lesson Level Data Base 59
Course Syllabi Segment Level Data Base 59
Existing Materials Data Base 59
Lesson Validation Data Base 59
Computer-Managed Instruction Data Base 59
Computer-Assisted Instruction Data Base 59
Lesson Specifications and Text Data Base 60

IV PROGRAM DEVELOPMENT AND IMPLEMENTATION PLAN 61

Program Priorities in Light of the Cost/Benefits Analysis 61
Benefits of Automated Support to FAISD Tasks 61
Cost-Related Development Priorities 66
Development and Implementation Resource Requirements 78
Hardware/Software Resources 79
Program Development Costs 79

V CONCLUSIONS AND RECOMMENDATIONS 81
Conclusions 81
Data Item 0001--List of Candidate ISD Tasks 81
Data Item 0002--Process Analysis of FAISD 81
Data Item 0003--Cost/Manpower Analysis of the ISD Process 82
Data Item 0004--ISD Task Priorities for CATSDM 82
Data Item 0005--Analysis of Existing Programs for CATSDM 82
VTS 83
Recommendations 83
Recommendation 1 83
Recommendation 2 83
Recommendation 3 83
Recommendation 4 84
Recommendation 5 84
Recommendation 6 84
Recommendation 7 84
84

Section Page

APPENDIX A.................................... 85
GLOSSARY...................................... 113

LIST OF ILLUSTRATIONS

Figure Page

1 CATSDM System of Computer Programs - General FLOW Diagram....................... 69
2 Correspondence Between Prioritized FAISD Model Tasks and CATSDM Programs............ 71
3 Phasing and Tasking Chart for CATSDM Development.. 73

LIST OF TABLES

Table Page

1 Summary of CATSDM Data Bases................. 50

SECTION I

INTRODUCTION

PURPOSE OF THIS DOCUMENT

This document is intended to provide the reader with a clear description of a model for Computer Aided Training System Development and Management (CATSDM) and is designed to serve as a functional specification base from which more detailed design and development activities could proceed.

In previous documents prepared as part of the Computer Aided Training System Development and Management (CATSDM) to ISD (Instructional Systems Development) project, a series of analyses were conducted to determine the characteristics of an idealized computer hardware/software system which could support the Instructional System Development (ISD) process. Analyses were also conducted to prioritize support needs for the various technical ISD processes. Finally, existing relevant hardware/software systems were examined to determine their general capabilities and potentials in relationship to the identified requirements.

Using the analyses outlined above, a set of functional specifications has been prepared describing a system of computer programs to assist in successfully and reliably carrying out the ISD process. These programs are designed to lend structure to the activities of persons involved in ISD, as well as to provide a wide range of support functions, including needed data collection and manipulation capabilities for all phases of the process.

OVERVIEW OF THE CONTENTS

The remainder of this document will deal in detail with the components of CATSDM, conception and development of those components, and recommendations and conclusions made in light of the analysis of existing systems, including Navy Training device number 11869, referred to in this document as the Versatile Training System (VTS). This will not only provide the reader with a feel for what CATSDM is and does, but also with how the benefits of CATSDM can be realized and integrated with what is already available through VTS.

The section on "Description of Methods" immediately following this section will describe the general procedures followed in developing this data item. It will outline how the VTS was studied, analyzed, and compared to CATSDM.

Section II will contain three subsections which will 1) overview the recommended system of programs, 2) describe in detail program capabilities (both standard across the whole CATSDM system and specific to individual subprograms) and, 3) describe the data bases created and used by the programs.

Each of the detailed descriptions of program capabilities will be accompanied by a table listing what types of information will be entered into each data base, and specifying how it was entered. A large portion of the capabilities available for one program will also be available in other programs. Therefore, instead of describing each capability over and over as a part of each program, a single set of descriptions of these common capabilities will be presented separate from individual program descriptions. By describing the standard capabilities in detail once, it will then be reasonable to merely reference them in the descriptions of each of the subsequent programs.

The description of the data bases included in the third subsection will include a series of tables outlining the projected contents of each of the data bases. For each piece of data in each data base, the program used to enter that data will be identified, and the programs which access that data will be listed.

Analysis of the VTS in light of CATSDM recommendations will be provided in Section III. The purpose of this section is to evaluate the VTS to determine how its capabilities relate to requirements of the proposed CATSDM system. The analysis will outline where the VTS address the CATSDM need as specified, and where it takes an alternative but acceptable approach to accomplishing each specific CATSDM task. It will also identify where the VTS does not now provide the required capabilities in any useful form.

Section IV will present a recommended plan for how to go about the development and implementation of the proposed CATSDM system. It will specifically address how the VTS could be systematically expanded and modified to meet the CATSDM requirements. The plan will attempt to organize development tasks in such a way that individual programs and subsystems can be developed one at a time and in such a way that each will provide a useful product in and of itself. In this way many of the benefits of CATSDM can be realized very early in the development effort, without having to wait for the whole system to be completed. The section will also present development tasks, phasing, and staffing.

Section V of the report will summarize all conclusions and recommendations.

DESCRIPTION OF METHODS

This section will describe the general procedures which were followed in developing this data item. It will outline how the VTS was studied, analyzed, and compared to CATSDM.

This data item was prepared from several major data sources. The first obviously was the preliminary work done on the previous data items and their extension into the description of programs, data bases, and so on required for CATSDM.

Information on the Versatile Training System (VTS) was gathered from two major sources. The first was a set of the best existing written documentation materials available on the VTS.

Specifically these are:

VTS & CATSDM - Outline of CATSDM VTS Overview CATSDM Explanations

VTS - Vol. I - FRAMP User's Manual No. 1

VTS - Vol. II - FRAMP User's Manual No. 2

VTS - Vol. III - FRS Aircrew Personnel User's Manual

VTS - Vol. IV - FRS Aircrew Personnel User's Manual

VTS - Vol. V - Subsystem Specs for Aircrew and Enlisted Men

- VTS Functional Description for Naval Aviation Activities

The general approach taken to utilize this documentation was to overview all the written materials to orient the contractor staff to systems capabilities and the documentation formats and layouts used. Then, utilizing the description of programs and standard capabilities implied by CATSDM, the documentation was searched specifically for VTS implications to each section of the CATSDM model on a program by program and capability by capability basis.

Finally, in those areas where the written documentation was inadequate (the documentation for VTS is not yet complete) follow-up phone calls and additional visits were made to NWC China Lake, where questions were clarified as much as possible. In those cases where information was unavailable or incomplete, assumptions were made in the anaysis of VTS/CATSDM match ups and were clearly identified as assumptions in this report.

The CATSDM functional specifications themselves were the joint product of a group of contractor personnel with extensive experience in large scale ISD (including NAVAIR, Army, Marines, and Air Force projects), in writing specifications for computer based training and training support systems, and in the operational development and use of such systems. In some cases the degree of detail provided in the description and definition of the CATSDM programs and their relationship to each other (See Section II) may suggest that this document might serve as more than a functional specification. That is not the case! Systems-subsystem specification, database specifications, and program specifications all require more detail than is given or intended here. They depend extensively upon the hardware/software environment selected for implementation.

The level of detail provided in this report is intended to be helpful and suggestive and not limiting or constraining. The relationships implied suggest functional, not structual definitions.

SECTION II

DESCRIPTION OF PROGRAMS

OVERVIEW OF THE SYSTEM OF PROGRAMS

In as much as the ISD process is an attempt at a systematic solution to complex training problems, CATSDM is designed to provide systematic structuring of the activities of persons, resources, and data involved in all phases of the process. Furthermore, CATSDM provides a system of computer programs which assists in the creation, manipulation, and updating of data bases critical to the ISD process. This section presents an overview of those programs, their interrelationships, and their associated data bases. It should be made clear here that these "programs" are really complex sub-systems in their own right.

Support is provided by CATSDM to both main line ISD activities clearly represented in the five ISD phases (analysis, design, development, implementation, and evaluation), and related peripheral activities (trainer requirements/procurement, tactics package development, and contracting concerns). The programs available through CATSDM are best understood when related to these groups of activities. In order to view program interrelationships more easily, we suggest following the flow diagram in Figure 1 as each of the programs is discussed.

Analysis Phase Programs and Data Bases

The Problem Analysis Program (PAP) assists in the first step of the analysis phase by helping to specify and interpret the kinds of data collected, and helping to determine the need for developing or revising an instructional program. In so doing, a problem analysis data base accessed by subsequent analysis phase programs is created. This data base also contributes to the Master Plan Program (MPP) and the Procurement Package Program (PPP), both of which are described later in this overview.

The Task Listing Program (TLP) assists in the creation of a complete set of tasks performed in the job or jobs being analyzed. In addition, the tasks are structured and numbered so superordinate/subordinate relationships are clearly shown and can be easily manipulated (inserted, deleted, revised) without cumbersome manual renumbering, retyping, etc.

Parallel to task list development a determination of student entry level is made using the Student Entry-Level Program (SELP). This establishes a student entry level data base and means for interpreting that data.

Student entry level data and task listing data bases are utilized by the Task Validation and Selection Program (TVSP), to assist in verifying the task list for completeness and accuracy and in selecting tasks for which training is required. As a result, the task list data base is updated as necessary and tasks requiring training are earmarked.

The Objectives Hierarchy Program (OHP) then assists instructional developers in establishing a hierarchy of learning objectives based on the task list. The resulting objectives hierarchy data base shows relationships among objectives and eventually suggests the order in which they should be learned. This data base also provides the foundation for activities in the ISD design phase.

As objective hierarchies are created, one additional data base in the analysis phase is established through the Existing Materials Evaluation Program (EMEP). This helps specify which of already available instructional materials are suitable for teaching any objectives identified in the objectives hierarchy. These materials are numbered in an existing materials data base and cross-referenced to the objectives hierarchy data base.

Design Phase Programs and Data Bases

For teaching each objective identified in the OHP, the Media Selection Program (MSP) provides an interactive process for determining a set of acceptable media alternatives. These include any trainer hardware identified in the Procurement of Trainers Program (PTP), as specified in the Trainer Requirements Program (TRP). These two programs are overviewed in the section on programs peripheral to mainstream ISD activities.

It is then possible to utilize the Syllabus Development Program (SDP) to structure the syllabus development process, transfer appropriate information from the objectives hierarchy data base, and number and reference syllabus elements. The result is three data bases: lesson level syllabus information, segment-level syllabus information, and a lesson interrelationship matrix.

The Training Support Requirements Program (TSRP) assists the instructional developers in specifying and manipulating resource requirements data. The output of the program is an analysis providing for alternative media mixes and needed alterations to basic resource requirement tables, which are retained for future use in the TSRP data base.

At this point, the help of the Lesson Specification Program (LSP) can be enlisted in the development of individual design guides which are used later to create each actual lesson. Reference is made to necessary technical manuals, other related documentation, and tactics package information. Structuring and tracking of the process is also established.

The Implementation Plan Program (IPP) facilitates another important part of the design phase. This program assists in coordinating resources (both personnel and material) and in developing a complete description of how the instructional program is to be administered. All information is retained in an implementation plan data base and is updated as the program implementation date gets closer.

Finally, the Quality Control Program (QCP) aids developers in establishing a complete tracking and evaluation procedure for all aspects of the developmental process. It helps specify the kinds of data to be collected and analyzed. It also helps pinpoint the persons needing the data and the time they need it. The QCP creates its own data base as the process is carried out.

Development Phase Programs and Data Bases

With the assistance of the Lesson Authoring Program (LAP), previously created lesson specifications are developed into complete lessons. The LAP provides help in formatting textual components of all lessons and allows for direct transfer of specifications to the lessons in the lesson specifications and text data base.

Authored lessons can then be produced in ready-for-paste-up copy (or directly transmitted to a phototypesetter) with the aid of the Lesson Production Program (LPP). In cases where audiovisual media are involved, this program assists in producing working copies of scripts for production personnel. In addition it can be used to track progress and staffing. All lessons will be stored in the LPP data base for easy on-line revision as needed.

Once produced in a usable form, the lessons are given to typical sample students to test for instructional effectiveness. Student performance, attitudinal data, and reviewer comments are collected, cataloged, and analyzed through use of the Lesson Tryout Program (LTP). The LTP also assists in the formulation of revision specifications based on the data collected. All of this information becomes part of the lesson validation data base.

Implementation Phase Programs and Data Bases

In the implementation phase a Computer-Managed Instruction Program (CMIP) can be used to administer student tests on-line, to diagnose student deficiencies from test results, and to prescribe student instructional activities. This program can also take charge of maintaining student records (i.e., a student's progress in the course can be assessed at any time). In addition, assets (personnel, materials, students) are scheduled in such a way as to assure that instruction is administered in an efficient and effective manner.

A Computer-Assisted Instruction Program (CAIP) can carry the implementation a step further by actually administering instruction on-line under computer controlled interactions.

Programs Peripheral to Mainstream ISD Activities

Three programs are geared to deal with instructional situations requiring complex trainer/simulator equipment. They are the:

1. Historical Trainer Data Program (HTDP), which collects and retains pertinent characteristics and requirements data of previously used trainers and allows for comparison with current needs.

2. Trainer Requirements Program (TRP), which assists in establishing requirements for trainers, along with utilization estimates.

3. Procurement of Trainers Program (PTP), which provides for contracting arrangements for new trainer equipment or repair or maintenance of existing trainers.

Each creates its own data base. All three input to the Media Selection Program (MSP) in the design phase.

A Tactics Package Program (TPP) provides for the structure and organization of the tactics package. Tactical doctrine is referenced in the objectives hierarchy development activity by the OHDB. The tactics package is retained as a data base.

Concurrently with the early analysis phase activity, the Master Plan Program (MPP) assists in organizing the long-range project schedule. A time line, project assignments list, and PERT chart are created. All are part of the MPP data base and are to be reviewed and updated as needed.

Information in the MPP data base is input to the Procurement Package Program (PPP). This program assists in determining the need for, and contracting the services of, outside agencies for particular portions of the ISD process. A PPP data base is established and maintained.

A Navy Training Plan Program (NTPP) assists in setting up required NTPP formats and information to be included. The created NTPP data base will be accessible for on-line review and update.

A Progress Monitoring Program (PMP) is designed to track resource expenditures on a task-by-task basis throughout the entire ISD process. Time and capital used on each task are the key resources to be monitored and will be entered into a project management data base.

DESCRIPTION OF PROGRAM CAPABILITIES

In this section of the report, a more detailed description of each proposed program will be presented. Each description will be accompanied by a table listing the types of information to be entered into each data base, and specifying the mode of its entry. A large portion of the capabilities available in one program will also be available in other programs. Therefore, instead of describing each capability again and again as a part of each program, a single set of these descriptions will be presented.

Definitions of Standard Capabilities

The purpose of this section is to describe the capabilities available in a number of the proposed programs. By describing the capabilities in detail here, it will be possible to merely reference them in the descriptions of each of the programs. In some cases, description of a capability presented in this section will be in general terms. The exact definition of the capability in relation to a particular program will be explained in detail in the description of that program.

Interactive Protocols (Menus, Prompts, Cues). The system should be designed for ease of use by the most unsophisticated user. Therefore, the system should rely, wherever possible, on routine interactive protocols which cue or prompt the user in his or her interaction. For example, user interaction with the system should be prompted by easily understood cues such as "Type the conditions for Objective 1.3.4.6."

The system should allow users to select system options from an on-line menu whenever possible, as opposed to expecting the user to type the desired option from memory. For instance, the system might display a menu such as:

"Do you wish to

1. write a message to another user?
2. write a message to all users?
3. read your messages?

Please type the appropriate number."

This reduces the typing the user must do, makes the user more aware of system options, and reduces the user's memory load (i.e., users do not need to be able to type well or remember complex code words or syntax for using the wide variety of system programs available).

Advice. When the system is designed, it should be kept in mind that the users may be relatively unsophisticated both in the use of computers and in the process and products involved in the ISD approach.

In order to further reduce the memory load the training needed by users, the system should provide an on-line "advice" capability. "Advice" should be available on two levels. On one level the system should provide basic information on how to manipulate the system. The information given should be as specific as possible to the particular program being used. For instance, if the user is using the review/comment function, the "advice" function (when requested by the user) should inform the user how to "asterisk" errors, how to record comments, how to move on when a comment is completed, how to exit from the program, etc. On the other level the system should explain terms, define input, provide information as to possible sources of data needed as basic input, and give basic training as to what the specific program being used does. For example, if a user is given the cue "type the conditions for Objective, 1.3.4.6," the user should, upon seeking advice, be given a clear and specific explanation of "conditions" and any related information as to standard format to be used. If other information relating to "conditions" is contained in some off line document, the "advice" capability would reference this source material. (The "Library" capability could further assist the user in locating the document.) This level of advice can be used in the basic training of instructional developers (perhaps reducing their training time) and provides some quality control on the data input.

The "advice" available should be provided upon request by a simple command (such as a key labeled "advice" or "help" or "info"). In this way, experienced users can bypass repeated reading of lengthy cues, prompts, etc., and improve the efficiency of their work. Specific portions of advice should appear unsolicited when a user makes a detectable error in input, as part of the error reduction/detection program.

Library. The library capability of the system should provide a catalog of pertinent off-line source materials, and assist the user in locating the necessary materials or personnel. This function will be especially useful in those cases where historical data comprises a significant part of the data base.

The library should catalog materials which relate to the ISD process; copies of previously accepted reports, of student lessons, etc.; historical data on projects similar to the one undertaken, etc. It could also catalog personnel who qualify as experts in some specific area.

In order to facilitate finding the appropriate source material the library should catalog all sources in several ways--by topic (such as "media selection" or "quality control plan"), by title, author, and/or project. When possible, a synopsis of the material should be available on-line. Materials should be cross-referenced, and all entries should guide users to related topics, documents and sources. The library should also include all necessary information to "lay your hands on" the desired source material or to contact the appropriate personnel.

Error Detection/Reduction Routines. The system will, whenever possible, provide checks on the data input. In this way there will be a certain quality control of all data bases. Most commonly the system will check for range and domain type errors. For example, if the only possible legal response to a certain question is a number, the user might find that the only operable keys on the terminal keyboard are the number keys. All other keys would be locked out. In this way the answer will be at least within the appropriate domain. If a tolerance can be specified for acceptable inputs, this too can be checked. For instance, if all inputs should be two digits, no three-digit responses will be accepted. In cases where it is not feasible to lock out parts of the keyboard or limit the input field, the system will provide error messages and/or cues to prompt immediate correction of errors, whenever an analysis of user input can detect and classify an error.

In those data bases where specific syntax and format are critical elements, the system should make reasonable checks on these factors. For instance if a date must be written as December 1, 1977 and is input as Dec. 1, 1977 the system should note this error and either automatically convert "Dec." to "December" or alert the user to make the necessary change. The system should automatically check that no non-month word was used and that no illegal day number (i.e., December 45, 1977) was input.

Automatic Input of Data From Previous Inputs. The various programs within the system will be cross-referenced and linked in such a way that any specific piece of data need only be input one time. If that same piece of data is a necessary and predictable part of the data base of another program, the system will automatically retrieve that data piece and use it as part of the new data base. For instance, the objective of any given piece of instruction will be an important piece of data in several programs. Once the objective has been input into one program, the system will automatically retrieve the necessary objectives, use them as part of the new data base, and display them as needed in any and all reports.

Where such an automatic and predictable cross-reference is not possible, the system should offer a range of support functions (copy, edit, move, etc.) to help eliminate redundancy of manual effort once needed data exists somewhere (anywhere) in the system.

Structured Input. The system will define the data to be input, assist the users in collecting and organizing the data, and will structure the data bases into easily used tables, charts, lists, and reports.

The user will be assisted in collecting and organizing the necessary data in one of two ways.

1. Where the data to be collected is consistent across nearly all content, form-driven programs can be written. Printed paper forms can be provided for ease in collecting the necessary data. Clearly spelled out procedures will be given as to how to correctly fill out the paper form when collecting data in the field. When the paper form is correctly and completely filled out, all necessary data will be available for input. Input into the system will require merely replicating the information on the paper forms. In some cases forms can be machine readable. The paper forms

should be easy to use and all instructions should be clearly specified (either on the form or in an accompanying job aid) as to information to be collected and the correct syntax to be used. Syntax and format of the screen display and the definition of acceptable input into the system should be as similar as possible to the format and syntax used by the paper forms.

2. Where the type of data to be collected varies considerably across different content, interactive, branching programs will be written. In these cases, input of a specific piece of data will cue the system to request input of the appropriate next piece of data. For instance, if a user specifies that an objective has been developed in workbook format, the system might request data such as the number of pages. On the other hand, for another objective taught by slide/tape, the system might request length of presentation time in minutes and number of slides produced. Once again, clear-cut guidelines as to the procedures to be used in collecting the necessary data and/or criteria for making decisions will be provided.

Input from either form-driven data bases or interactive, branching programs will be structured by the system into various usable data bases which will be available to the user in the form of tables, charts, lists, and reports viewable on-line or in hardcopy printouts.

Automatic Numbering and Referencing. The system will, when appropriate, define the numbering and referencing system to be used. For example, objectives hierarchies would use the standard military system where Objective 1.1 is subordinate to Objective 1, and Objective 1.1.1 is then subordinate to 1.1, etc. All syllabi would use the standard unit number, lesson number, segment number. By defining the numbering and referencing systems to be used and requesting certain data in a predefined sequence, many inconsistencies and inaccuracies in references can be avoided. For example, the system would prohibit input of two objectives with the same reference number. It could be programmed to check and prompt users that segments listed in objectives hierarchies as prerequisite to other instruction should be taught in an appropriate sequence, (i.e., not in an order contrary to that indicated by the objectives hierarchy).

In large data bases, the renumbering of the entries caused by deletions and insertions can be very burdensome. Such changes should be provided automatically by the system. For example, if the syllabus for a given lesson indicates in a given lesson that there are four segments and it is decided to insert a new segment (*) between Segments 1 and

2, (1, *, 2, 3, 4,) leaving the remaining sequence the same,the "old" Segments should be automatically renumbered (i.e., "old" segment 2 becomes 3, 3 becomes 4, etc.)

The system should also provide appropriate linkage and cross-referencing of data. For instance, a given objective may appear as a prerequisite or enabling objective in more than one hierarchy and, therefore, have more than one hierarchy number. The same objective would be given different numbers in the final syllabus. The system must cross-reference and link identical data found in different data bases. In addition, the specification of identifiable subsets of data elements through general identifiers (i.e., 1.1.*.6) should be possible. This example given might mean to select every four field entry beginning with 1.1., with any legal entry in the third field, and ending with .6.

Hierarchy of Users. The system will allow a hierarchy of users to be specified [i.e., certain users can be prohibited from using certain system capabilities; specified users can be limited to the use of certain capabilities or to certain programs; and input can be tagged as to the user (or authority of user) making the input]. For instance, the review/comment capability might be restricted only to those authorized to review the report being dealt with. The on-line update capability would be further restricted to the use of those authorized to approve changes and knowledgeable in the appropriate procedures for making changes.

On-line Review/Comment. This capability will allow authorized users to review on-line (i.e., at a terminal) the data input or reports generated by the related program. Authorized users through the terminal keyboard could then "asterisk" or indicate in some standardized way areas where they deem that changes are necessary or problems arise. A mechanism will be provided whereby the user can then "comment" on each "asterisk" (i.e., explain specifically the problem identified, propose solutions, etc).

The "comments" could be left anonymous or require a signature. The signature could also be left to the user's discretion. Comments could be ranked according to the user making them (i.e., the comments of the person charged with the responsibility for review of that program could be tagged in some way so as to quickly distinguish these comments from those of other users).

Comments could also be categorized according to the type of error or problem identified. For example, after inputting an asterisk, the user might be asked to classify the error as

1. typographical or spelling error.
2. wording change.
3. content change.
4. other.

In this way, spelling errors could be quickly identified and changed without further ado, while suggested content changes could be reserved until any necessary approval for such changes had been received. A mechanism would be provided whereby those who were so authorized could review and comment, and indicate approval or disapproval of suggested changes made by others.

The review/comment capability would not allow users to permanently change any data input or any information included in a program report. At this time all comments would be considered suggestions, although some comments could be routinely accepted for change either because the error was routine (a spelling error requiring no approval), the user making the comment had the authority to authorize the change, or the comment had been approved by another user with authority to authorize changes.

On-Line Update. This capability would allow authorized users to make permanent changes in data previously input to the system. All text editing/word processing capabilities would be available in making updates and corrections. Where a single piece of data (such as an objective) appears as part of several data bases, a change in the input to one data base should automatically change that piece of data in all other data bases, unless those data bases contain modifications of the basic data. In this case the modifications are called to the user's attention for decisions and actions. For instance, if the wording of an objective is to be slightly changed in one data base, the system should automatically and consistently make the same change in other data bases in which the objective appears. All objectives needing a similar change should be readily located and reviewed by key word searches. Where a change in a single piece of data implies a consistent and specific change throughout a data base, the system should automatically make the change.

The system could also keep certain records useful in quality control and accountability procedures. For instance, it could keep track of the dates on which a data change was made, who made it, recommended it, authorized it, etc.

"What If...?" An extremely important and useful feature of the system described should be the ability to protect actual data bases and store multiple copies thereof as temporary files which can then be manipulated and changed at will. This function allows the user to play "what if" games. These "management models" are extremely important in making accurate projections or complex decisions on time, budget, personnel, and other management and instructional concerns. For example, the actual course being developed might have 40 videotape objectives specified. If it is found that the project is running overbudget, one alternative to change might be some of the videotapes to the second ranked media choice, slide/tapes. Using a temporary file, a "what if" data base could be established showing all videotapes as slide/tapes. The computer could then project (using data input regarding personnel needs, cost, time requirements, etc., for developing slide/tapes) what financial savings could be affected by changing these objectives to slide/tapes, what personnel changes would be needed, when production could be completed, etc. The on-line "what if" capability allows faster and more objective evaluation of alternative decisions than a manual evaluation.

Communications Facility. The system should have functions permitting the users to communicate with each other. Functions should be designed in such a way that one user can communicate with any other specified user, a subset of users (such as all Instructional Developers), or with all users. The system should allow storage of messages in such a way as to prompt users not currently on-line to check their messages whenever they log-on and reply or respond in whatever way is appropriate. This function should help in efficient production and save considerable time. For instance, an authorized user who has completed a review of certain material could notify the appropriate personnel of the material reviewed and in this way the material can enter the next phase of production as quickly as possible.

Structured Report Format. When the format, headings, and/or wording to be used in a formal report can be specified, the system will display the format to be used on the screen and leave blanks for specific pieces of data to be input. The user will only be required to supply the specific missing data.

This function could be written for two levels of users:

1. Unsophisticated or inexperienced users.

In this version of the structured report format, the system could number each "blank" left in the report and users could indicate by number which blanks they were ready to fill in. The user would then receive a number of cues as to the specific information belonging in the blank and as to the correct format to use. For instance, blank Number 1 might lead the user to a cue such as . . .

> "On what day is the report to be submitted? Supply the date in this form: December 1, 1977"

It is implied, therefore, that the program will automatically check for this format, no non-month word would be accepted, and no illegal day number (i.e., December 45, 1977) would be allowed. Where whole sections of the report are to be supplied, the instructions would necessarily be more general, but would provide as much guidance as possible. For instance, the instructions to the user might be:

> "This section should explain in general the rationale for the project plan being proposed. Usually the section is four to five paragraphs long and deals with the personnel needs of the training program, the availability of the personnel needed, proposed cost, and projected improvements."

When a user has specified any needed information, the system would insert the new data into the appropriate position in the report (and incidentally, in any other data bases where it might be needed).

2. Sophisticated or experienced users.

This version of the structured report format would allow the user to bypass cues about needed content and format and to supply only the needed pieces of information. The user could type the needed information into the report itself. The user would need to be provided with simple mechanisms for requesting additional space, for moving back and forth through the report, and for other editing activities.

Any user should be able to move easily from one usage level to another, as he or she might be experienced at writing one part of a report and inexperienced at writing another. In either version, users should be able to see at any point how the data input appears in the structured report format and revise as necessary.

Reading Grade Level Computation. Where sections of reports or actual instructional materials are to be kept on-line, a function should be provided to assess the reading level of any designated text passage. Several good algorithms exist for this function, and one should be adapted rather than reinvented.

This function should be able to analyze any identifiable text field on the system and return an estimated reading level in real time. "Advice" or "help" for this function, in addition to specifying how to use it, would briefly describe how it works and would offer a few helpful hints in reducing the reading difficulty of a text passage.

Text Editing/Curriculum File Maintenance Support. This capability would allow users to easily and efficiently correct and change data presently or previously input into the system. It should be designed for simplicity of use and should require minimal training for effective use.

Some of the capabilities of this function include:

1. Request for margins to be automatically right and/or left justified.
2. Ability to scroll forward or backward through a structured report.
3. Automatic page numbering of structured reports.
4. Automatic line count for page breaking of structured reports.
5. Simple commands for inserting additional text, deleting text, eliminating unnecessary blanks (close-up), rearranging blocks of previously typed text, and automatic reformatting of subsequent text.
6. Specification, storage, and easy retrieval of often used words, phrases, and paragraphs to prevent the need for typing the same text repeatedly in different context (copy/duplicate).
7. Specification of and automatic changes in tabs, indentation, and margin adjustments.
8. Automatic centering (both horizontal and vertical), and left and right justification.
9. Automatic decimal alignment.
10. Automatic search and/or replace for strings or automatic stopping at formatted input points.

11. Easy commands for indicating color codes, changes in font (script, boldface, superscript, subscript), or other text attributes.

12. Automatic underlining, uppercase.

Hard-Copy Printout. This capability will allow the user to request a formatted paper copy of a specific data base or report generated by the CATSDM system. Such paper copies are often useful for review purposes and provide valuable historical records.

Hardcopy printouts could be of three basic types:

1. A hardcopy printout of a single screen display.

Any display or part of a display appearing on the computer display screen could be printed. The use would not need to spend time taking copious "notes" on information displayed.

2. A listing from a data base.

Once all related data bases had been input, the user could get a structured, easily understood printout of a specific data base. For instance, once all pertinent information related to media selection had been input, a user could request a listing of all objectives using videotape. A series of menus could allow the user to specify the specific subset listing required.

3. A printed, structured report format/report.

When a report format can be specified, the system can provide a printed version of the format, supplying all standardized wording, divisions, etc., leaving blank areas for any specific information to be supplied. Once any pertinent data to be included in the report is supplied to the system, a printed version of the report, including the new data, can be requested.

Progress Tracking. The progress tracking function of the system is an on-line management tool. It allows users to break major tasks (such as developing a final Navy Training Plan) into subtasks, to assign each subtask to different personnel, establish due dates, and track progress (noting such things as who has reviewed the data input, who has revised it, when these tasks were performed, etc.). This capability will facilitate lower-level management functions where it is less likely that professional or experienced managers will be available.

Standard Data Base Management Functions. The CATSDM system used needs to have a data base management system which will include such attributes as keyword searches, indexed access, nonredundancy of data, data security at various levels, data base creation, data base updating, inquiry, and report generation. It will provide for field validation on entry, update and error recovery, and sorting of the data base. Standard CODASYL specifications and standards for Data Base Management should be followed to the maximum extent possible, in order to increase the level of standardization between systems.

Description of Functions of Each Program

This section of the report will describe in some detail the purpose and function of each program in the CATSDM system. Each description will be supported by a table listing the exact types of information to be collected as part of the program, how it will be manipulated, and in what data base it will be stored. For organizational purposes these tables are linked in this document directly to the appropriate data base description, and so appear in the next section with the data bases.

Analysis Phase Programs

Problem Analysis Program (PAP). The Problem Analysis Program will assist the user in the collection and structuring of the information necessary to determine the program goals, constraints, resources, and problems which will shape the ISD process. The program will prompt the user to gather all the categories of information needed (such as mission characteristics, weapons system design characteristics, existing training facilities and equipment, existing schedules, funding levels, and required end product characteristics). The system will also provide advice as to possible sources of such data.

The program will allow on-line storage of the data and will organize and structure it so that it can be most easily retrieved and analyzed by the user.

The program will need constant updating and should therefore have on-line review/comment and edit/update capabilities.

The final output of the program is a Problem Analysis Report. The system should support this effort by its structured report format and text editing/word processing capabilities. Previously developed Problem Analysis Reports should be available either on- or off-line as references and be easily located using Advice and Library system capabilities.

Task Listing Program (TLP). The Task Listing Program will assist the instructional developer in the analysis of a given job and in the development of an orderly list of the major tasks to be performed by an individual working in that job. The TLP will help organize the analysis procedure by on-line cueing and prompting and by ordering the input of the data base. The user would first be requested to specify all responsibility areas (or systems) in the job. The TLP would then have the user select one responsibility area and mentally walk through it, listing each mission (or subsystem) related to that area. The program would guide the user through the entire analysis procedure until all tasks performed under both normal and extraordinary conditions were identified. If at any point the user were unsure of the input expected, the TLP should provide an explanation of the task listing development process and definitions of terms, etc., upon request.

The program would automatically provide the standard numbering and referencing system familiar to the military community where Mission 1.1 is subordinate to Reponsibility Area 1; where Phase 1.1.1 is part of Mission 1.1, and so forth.

After all tasks are identified, the TLP will request the user to specify conditions and standards for each task in a standardized format.

The TLP will have full use of on-line review/comment and update capabilities. These will be especially useful during the validation phase of the task listing development process. The TLP will, therefore, need a full range of editing capabilities. One of the most important of these capabilities would be the system's automatic restructuring and renumbering of the tasks listed as tasks are deleted or added to the original listing. The program will require an automatic cross-referencing system based on keyword searches to permit easy identification of identical or related tasks falling under different responsibility areas.

Much of the data base input as part of this program will need to be tagged and automatically supplied as part of subsequent data bases for such programs as the Objectives Hierarchy Program (OHP).

Student Entry-Level Program (SELP). The Student Entry-Level Program will assist instructional developers in determining and describing the characteristics of students coming into the course(s). It will indicate what data types need be obtained and how to find them. It will also provide for structured input and data reporting.

Interactive protocols will be employed to help the user and to determine what student entry characteristics should be described. These may include previous related training (including terminal objectives mastered), aptitude indicators (such as GCT scores), related job experience that is required and/or can be substituted for training, and other useful data (such as age or pay grade). Much of this information will be available in the Problem Analysis Program (PAP) data base, which will be accessed through this program. In addition, the user will be given suggestions as to where to obtain the necessary data, and a checklist for tracking his or her efforts.

Structured input capability will ensure that any entry level data is entered correctly and numbered automatically. Terminal objectives from previous training will be numbered so that they can be cross-referenced as prerequisites to objectives in the course under development at the appropriate point in the development process. Entry level data will also be entered so developers can see the range of characteristics of the target audience, as well as what the typical or average student looks like. For example, it might be found that the average student has a score of 76 on a given aptitude test for which the student population scores range from 59 to 92. Other test parameters should be available as needed (e.g., mean, standard deviation, etc.).

Structured report formats will be available for creating hard copy printouts of entry level data (student entry level report) in a form that can be utilized in the development effort. The user will also have access to previously generated reports to review as examples.

Task Validation and Selection Program (TVSP). The Task Validation and Selection Program will assist the instructional developers in determining the validity of the tasks included in a task listing and in sorting those tasks into the following training categories:

1. No formal training.
2. Review training only.
3. Familiarization training only.
4. Formal training at FRS.
5. Training at operational squadron.

Once the instructional development team has determined the task selection algorithm and the exact questions to be asked during the task validation survey, the TVSP will assist in formatting and printing out the necessary survey forms by using the program's text manipulation and hard copy printout capabilities. The instructional development team will determine the appropriate groups to be surveyed, and the program will either (a) administer the survey on-line at all appropriate sites (prompting to allow maximum ease of use for what is most likely a group of inexperienced users), or (b) allow for batch input of survey data from appropriately designed and compeleted survey forms when the survey must be conducted in the field through use of paper/pencil forms with no access to computer terminals.

The TVSP will analyze and summarize the survey data, structuring the information into a collapsed, easily understood form, including a catalog and summary of comments made by those surveyed.

Once the validation survey data has been input, the TVSP will assist in the preliminary selection of tasks for training and sort them into the categories described above. In order to make such selections, the TVSP will use the task selection algorithm developed by the instructional development team, the survey data, and any necessary data from the Task Listing Program (TLP) and the Student Entry Level Program (SELP).

The selection of tasks done automatically by the TVSP will be a preliminary or "rough cut" of the final selection of tasks. The instructional development team will make refinements on line by using the on-line review/comment function. Authorized users will make permanent changes by using the editing and update capabilities.

Objectives Hierarchy Program (OHP). The Objectives Hierarchy Program will assist the instructional developers in analyzing the tasks selected for inclusion in the training program and in developing the required objectives and hierarchy diagrams.

The OHP will provide on-line protocols to guide the instructional developers in listing, in a logical and orderly fashion (with advice upon request), all component subtasks, including decision component and recall component subtasks, for each major task selected for training. Using the various editing functions, the user should be able to add, delete, or consolidate subtasks with relative ease. The program will automatically provide and revise as necessary the numbering and referencing of subtasks, through use of the standard military referencing system. Authorized users should be able to review/comment on the list of subtasks produced.

The OHP will also provide a means for input of instructional objectives for each task (behavior to be performed, conditions of performance and minimum acceptable performance standard).

The program will automatically tag each objective input with a reference number by using the standard military referencing system. This system indicates the subordinate, superordinate, and independent relationships existing between the various component subtasks (Task 2.1.1 is subordinate to Task 2.1, etc). Tasks and subtasks appearing more than once in the hierarchy will show alternate numbers to point out the existing duplication and its place of occurance.

Given the data stated above, the program will produce the necessary objectives hierarchy diagrams for the course being developed. It will also provide a catalog of information related to each objective to be trained, including necessary cross-references to the original Task Listing, the Task Listing Validation, and the Task Selection Report. The program should provide an automatic search and cross-referencing system to enable the instructional developers to easily find identical subtasks appearing in different parts of the objectives hierarchy. The program should automatically tag such objectives on the output diagrams as well as in the catalog.

To accommodate the necessary changes and updates to the objectives hierarchy diagrams, a full range of review/comment, edit/revise, and manual override of program inputs and outputs is needed.

Existing Materials Evaluation Program (EMEP). The Existing Materials Evaluation Program assists instructional developers in collecting, evaluating, and cataloging existing instructional materials for possible use in the course under development. The result of this activity is a list of appropriate materials cross-referenced to objectives in the objectives hierarchy.

It is first necessary to determine possible sources of existing related materials. The EMEP will display interactive protocols enabling the user to determine generic content areas addressed in the objectives hierarchy. It will then advise the user concerning publications, directives, and established training facilities that address training in those generic areas. This makes it possible to locate and request materials for review.

Once materials are obtained, their learning objectives can be compared to those in the objectives hierarchy data base for applicability. Suitable objectives can be entered into the existing materials (EM) data base and automatically numbered and cross-referenced to the objectives hierarchy. Information from the student entry level data base can also be accessed and compared with those characteristics of the existing materials to determine if such things as reading grade level are appropriate for the target population.

The EMEP will provide guidelines for evaluating the quality of existing materials. It will assist the user in analyzing available student performance data, correspondence of objectives to instructional strategies and evaluation criteria, and appropriateness of media used. These analyses could be performed using subroutines from other programs such as the Media Selection Program (MSP) and the Lesson Specification Program (LSP), or they may be based on existing validated instruments such as the Instructional Strategy Diagnostic Profile (ISDP).

When completed, the numbered and cross-referenced materials will be retained as the EM data base. By using the on-line update capability additional materials can be reviewed and added as necessary.

Design Phase Programs

Media Selection Program (MSP). The Media Selection Program will assist the instructional development team in making consistent and instructionally and economically defensible decisions on what type of media to use for teaching any given objective.

The instructional development team will examine the appropriate reports and documents to identify available media and will then develop an algorithm to sort the objectives into the various media categories. The Library capability will be useful in this effort. On-line interactive branching protocols will guide users through the media selection algorithm and provide the program with necessary input on whether each objective is a classroom (study session) type of objective or a hands-on (device session) type of objective, etc. Once the necessary priority decision rules are input, the MSP will identify the optimum delivery medium and all alternative acceptable media for each objective in the objectives hierarchies.

The MSP should also provide information regarding the number of objectives selected for each medium and provide lists of objectives and related information categorized by media.

When available instructional media are not clearly defined, the MSP should help specify the instructional requirements of "to-be-acquired" media by cataloging and summarizing selected responses to the media selection algorithm into a design report. For instance, if trainers have not yet been designed for a new weapons system, the MSP program should assist the trainer designers to determine what capabilities the trainers must have to best meet the training needs of future students. If a decision must be made as to whether workbooks need color or can be produced in less expensive black and white, the MSP should be able to point out those workbook objectives where color is instructionally critical and which alternatives are available, etc. Thus the decision made will be both instructionally and economically defensible.

Syllabus Development Program (SDP). The Syllabus Development Program should assist the instructional development team in developing lesson maps, ordinal course syllabi, and time-based syllabi (class schedules).

The SDP should provide on-line, interactive protocols which will structure the material and guide the instructional developers through the process of sequencing and organizing the objectives selected for training into a workable course outline. The program should assure that the final organization of the course allows students to (a) proceed from easier to more difficult material within a task, (b) have early hands-on experience whenever possible, and (c) address the most critical job tasks as early as possible. In addition, objectives should be grouped into lessons in such a way that (a) use of trainer-requiring lessons is made as efficient as possible, and (b) only one, or possibly two, presentation media will be needed within a single lesson. Data from the Objectives Hierarchy and Media Selection programs will be essential in determining that all necessary factors are considered (data on prerequisites, on appropriate media, etc., from the OHP and MSP should be readily available to the user if requested) so decisions made during the SDP are based on accurate data. The program should provide some sophisticated quality control by cross-referencing with the OHP and MSP. For example, the program should automatically check and cue users if a lesson map appearing in the course outline sequences objectives so as to violate the subordinate/superordinate relationship specified in the Objectives Hierarchy (i.e., so students do not receive necessary prerequisite objectives first) or if a lesson presentation medium is selected which is inappropriate for certain objectives included in the lesson.

The program should make extensive use of the automatic input function which will automatically copy significant pieces of information from the OHP and MSP data bases into the data base needed for the syllabus. For instance, the user should not need to retype the conditions, behaviors, or standards for any objective, as these were previously input. The program will also provide an automatic numbering (and renumbering) and reference system.

When all necessary data is input, the program should produce, in a specified format, lesson and unit maps (diagrams) indicating superordinate, subordinate, and independent relationships between objectives. It should also provide an appropriate ordinal sequence for progressing through the training program, as well as lists and/or catalogs of all related objectives and related numbers and references, suggested lesson media, time estimates for student completion, and other pertinent cross-referencing data.

Once the ordinal syllabus has been developed, the program will assist the user in developing a suggested daily class schedule for all training programs and overlapping sessions of each course. This will be done by interactive branching protocols and automatic retrieval of data from other data bases. This schedule will allow for systematic consideration of such factors as total syllabus hours, device-session hours, estimated number of students, number of devices available at a given time, etc. Once the pertinent data has been input, the system should help develop a time line for progressing through the training program (time-based syllabus layout) and a systematized way of developing a daily class schedule based on a specific calendar.

The "What if..." capability will be an essential part of the design of this program. This will allow the instructional development team to project the time, cost, personnel needs, etc., of a wide range of situations relating to development of the ordinal syllabi and class schedules. For instance, the program should identify periods of peak resource utilization. The team could also ask the program to project the effect of one trainer being "down" for two weeks for repair during various units of instruction, how much time would be lost, what alternate ordinal syllabus could students follow, etc. In this way, the least disruptive time could be allotted for routine repairs of trainers and so forth.

The program should also assist in preparation of the hard copy printed versions of syllabi to be used by developers, students, instructors, and managers and of other documents such as ordinal lists of objectives, etc.

Training Support Requirements Program (TSRP). The Training Support Requirements Program assists the instructional developers, management personnel, and those who will implement the final courses in specifying and manipulating resource requirements data.

Much of the data needed for this program can be supplied automatically from previously input programs (PAP,OHP,MSP,SDP). Analysis of data from these programs will permit estimates of needed personnel, equipment, facilities, and services for completion of design and development of the course materials. The program should cue the user to collect additional data (such as student throughput data, etc.) affecting the resource requirements for implementation and ongoing instruction. Similar data and estimates should be input regarding continuous evaluation, revision, and updating of the materials over time. Using all this data the program should assist planners and managers in determining possible allocations of total resources for completion and maintenance of the project throughout all remaining phases of ISD.

The TSRP should organize and structure the data input in such ways as to best facilitate the needed analyses. For instance, the program should create or structure basic resource requirements tables from the data input. On-line review/comment and edit/update capabilities are essential to ensure that data is kept current and accurate. "What if..." manipulations and projections are critical to this program so the effects of various media mix selections and other resource allocations can be predicted and the resource requirement tables altered accordingly.

The final output of the TSRP is a formal report suggesting alternative course designs with alternative media mixes and predictions of the total resources required for each alternative. This is used as a guide in the remaining phases of ISD so the project can select reasonable alternatives and will not collapse at any point for lack of resources.

Lesson Specification Program (LSP). The Lesson Specification Program will assist the subject matter experts and/or other instructional developers in developing the specifications used later by authors as a design guide in authoring student materials.

The LSP will consist of on-line, interactive, branching protocols which organize and structure the development of lesson specification documents and guide the developer in selecting and developing instructional components and strategies appropriate for each objective. The specific components and strategies requested by the program will depend upon the behavior level (use or remember) and the content classification of each objective. The following components will be necessary for most objectives (as called out in the FAISD model):

1. Complete objective (behavior, conditions, and standards).
2. Generality.
3. Description of the instructional help.
4. Description of the type and number of examples.
5. Description of the type and number of practice and test items.
6. Description of any special teaching points.
7. Description of graphic requirements.

Others may be included as needed. To make the system more generalizable, provisions could be made for user definition of all instructional components. In that case on-line protocols for directing and disciplining such input would be provided.

Through appropriate cues, prompts, and the on-line advice or help function the developer could then be assisted in correctly writing each component. An example of such a component would be an appropriate, complete, and technically correct generality to be used 1) by students as the core information to be learned and 2) as major input into the development of NATOPS, flight and technical manuals. The specifications writer could be further assisted in making appropriate decisions concerning a) the type of instructional help needed, b) the number of examples, practice and test items required, c) the specific example format and strategy to be used, and d) the practice/test format and strategy, etc. Once such decisions were made, they would be translated into an easily understood format for later use as design guides by authors when producing the final materials.

The program would structure and track the entire lesson specification development process to help assure implementation of proper review cycles necessary for adequate quality control and accountability. The program would keep status information on each objective being developed. It would also produce status reports indicating what portion of the lesson specification (LS) was completed, who had developed each portion of the LS, who had reviewed it or was responsible for its review and approval, what was the time taken for each stage of development and review, etc.

To ensure that the lessons to be developed are instructionally sound, lesson specifications must be reviewed at several points by instructional experts. To ensure that content is accurate and current, subject matter experts must also review all lesson specifications. Therefore the program will require a full range of review/comment and cross-referencing functions, renumbering, etc., to assure quick and easy implementation of quality control functions and updates.

The program will eventually produce (both on-line and in hardcopy form) complete lesson specification documents in a standardized format. The program should also produce formatted reports of prespecified data base subsets. For instance, it should be capable of producing a summary of each lesson, showing lesson map, segment titles and reference numbers, objectives and generalities for each segment within the lesson, media specifications, etc.

Implementation Plan Program (IPP). The Implementation Plan Program will assist project planners, managers, and those who will in the future implement and evaluate the instructional program in developing an implementation plan. The plan is used for coordinating resources and establishes the procedures to be applied in providing necessary training for instructors and instructional managers. The plan is also used for conducting instruction and evaluating students; for managing facilities, equipment, materials and personnel; for scheduling; for integrating the new course with existing courses; and for coordinating implementation with quality control efforts.

Much of the data needed by this program can be provided automatically from previously input data bases (such as TSRP). The IPP should cue and advise the user on collection of other pertinent data. The program will provide review/comment and update capabilities so the data may be constantly updated as the program implementation date gets closer.

The program needs "what if" manipulations so alternative implementation plans can be compared.

Word processing capabilities will be needed in order to produce initial, updated, and final implementation plan reports.

Quality Control Program (QCP). The Quality Control Program will assist the user in developing a report describing in detail a plan for complete and continuous assessment of the effectiveness, efficiency, and palatability of the instructional materials and management system.

Through the use of interactive protocols the program will help the user design the plan to be used for formative evaluation, including small group tryout procedures, determination of methods of measurement and variables to be measured, and identification of review points and procedures. These will be extended to include ongoing quality control during full implementation of validated training. Flowcharts and verbal descriptions of the plan will be needed.

The program will also assist in structuring external quality control procedures, which will ensure that training meets, and continues to meet, operational requirements for skilled personnel.

Sample data collection forms will be stored for use in the quality control process. On-line review and edit capability will enable users to modify forms as necessary to meet the needs of the instructional system.

The program will provide a job aid in the form of a checklist to assist users in collecting all data, doing the appropriate analyses, and developing the final Quality Control Plan report. The system will then assist in the production of the hardcopy of this report by permitting on-line entry, review/comment, edit etc.

The program will assist in the development of a quality control manual based on the quality control plan, which can be printed in hard copy and distributed to appropriate personnel. Through use of the on-line review and edit capability, the manual will be updated as needed.

The QCP data base will serve as input to the Lesson Tryout Program, which will actually collect and analyze student performance data based on QCP guidelines.

The QCP utilizes input for the Problem Analysis data base and the Master Plan Program data base.

Development Phase Programs

Lesson Authoring Program (LAP). The Lesson Authoring Program will guide and track 1) development of prototype versions of lessons to be used in small group tryouts, 2) lesson revisions on the basis of such tryout data, and 3) readying of the final version for production.

Using the data in the lesson specification and tactics package programs and the media format guide as blueprints or as direct input (as in the case of objectives), the LAP would guide the lesson authors in developing the specified instructional helps, examples, practice/test items, and other instructional components for a paper-and-pencil version suitable for initial small group tryouts. This initial version could be input directly on-line or handwritten and then entered on-line. The program would track the complete authoring process including all review and accountability cycles.

The program would also organize and structure the collection and analysis of data during the initial tryout of the prototypes, assisting the developers in deciding what revisions are necessary. Using the review/comment and edit functions, necessary revisions could be made to the prototypes and they could be prepared for entry into the final production cycle.

The final output of the LAP is a printed version of student lessons suitable for final production in the various media previously determined to be most appropriate. These printed versions must contain adequate descriptions of and references to all graphics (charts, drawings, photographs, slides, film, etc.) to allow the graphics production crew to produce the desired end product.

Differences in media imply differences in end products. (For example, a slide/tape lesson requires an audiotape, a set of slides, and a worksheet. On the other hand, a trainer exercise demands an instructor guide, a worksheet, and an evaluation checklist.) The printed form of the authored version must be formatted in such a way that final production into these different end products will be most efficiently carried out. This implies that different formats will be needed depending on the end product (e.g., authored workbooks going to a printer will be formatted differently than authored slide/tape worksheets, or authored storyboards for videotapes, or authored computer-assisted instructor lessons). All necessary technical information related to the media to be used must be included in the authored lesson. For example, for a slide/tape this means: How much lead time is necessary on the tape? Is a focusing frame necessary? What background music or sound effects are required? What about automatic advance tones, etc. The LAP will have guided and assisted the developers and production personnel in making these decisions through appropriate interactive branching protocols.

Lesson Production Program (LPP). The Lesson Production Program will receive input in the form of prototype lessons from the Lesson Authoring Program (LAP). The LPP will format printed portions of lessons and worksheets for paste-up, provide working copies of scripts for production efforts in audiovisual media, and format evaluation checklists, instructor guides, and other pieces of instruction. In addition, tracking and structuring of the production process will be done.

A key feature of this program will be the capability to review printed lessons on-line and insert necessary production codes for type font and formatting considerations. When linked to compatible typesetting equipment, it will be possible to print-out production-quality materials ready for paste-up or typeset in hard copy. It will also be possible to review and edit text and production codes as necessary on the system to accomplish revisions as specified in the Lesson Tryout Program (LTP).

Where lessons incorporate other than print media (slide/tapes, videotapes, etc.), working copies of scripts will be formatted and printed out in hard copy for production personnel. Worksheets, evaluation checklists, and instructor guides can be produced like printed lessons to accompany nonprinted materials. Review and edit capability will also be available for all these items, including scripts.

The entire production process will be tracked and structured to provide information on lesson production progress and revision and staffing of the production effort. In this way, maximum use of personnel and material production resources will be realized.

Lesson Tryout Program (LTP). The Lesson Tryout Program will consist of two relatively independent sections. The first will assist the user in developing a specific plan for lesson tryouts consistent with the quality control plan. The second will provide for input and analysis of reviewer comments and student performance and attitudinal data obtained during the tryouts. Output will consist of specifications for revision of materials based on tryout data.

The first section of the LTP will access the Lesson Production Program and Quality Control Plan data and assist the user in developing a tryout plan through interactive protocols. The plan will specify the extent and nature of individual and small group tryouts, including a lesson-by-lesson time line. On-line update capability will enable the user to revise the tryout plan as necessary in accordance with the actual production cycle.

A variety of suggested data collection forms will also be provided as output. These will consist of student attitude surveys, review comment forms, and additional forms as needed. Where possible, the forms will be machine readable for expeditious input of data (e.g., mark-sense).

The second section of the LTP will provide for entry of all tryout data including attitudinal data, performance data, and reviewer comments. Analyses of the data will be performed on-line and provide several outputs such as:

1. Item analysis indicating student performance on test items and resulting mastery of objectives.
2. Student attitudes toward the materials by objective.
3. Reviewer comments on materials by objective.

The results of the analyses performed will be used to assist the user in formulating revision specifications on-line. Interactive protocols based on the analyses will guide the process, objective-by-objective. Revision specifications will then be cross-referenced to Lesson Specification, Authoring, and Production Programs where they can be implemented. The formative analyses and revision specifications will also be organized using the structured report format capability and will be printed out in hard copy.

Implementation Phase Programs

Computer-Managed Instruction Program(s) (CMIP). A variety of CMIPs could be developed to meet the needs of the particular instructional system. They would include combinations of the following capabilities:

1. A variety of tests (pre, post, entry, and progress) could be administered on-line and scored automatically. If tests were not administered on-line, paper-and-pencil tests could be machine scanned, entered, and scored by using compatible machine readable answer sheets.
2. Student performance data could provide the basis for criterion-referenced diagnosis of strengths and deficiencies making up student feedback.
3. Students could be given prescriptions for remediation or further study based on diagnosis.

4. Complete student records could be maintained in real time, providing accurate and instantaneous hard copy printout of all or part of an individual student profile.

5. Banks of test items could be maintained and manipulated to provide random alternate forms of objective-based student tests.

6. Student performance data could be automatically combined to create large data bases, which could provide normative references and indications of instructional materials effectiveness.

7. Personnel and physical resources could be scheduled and allocated automatically in accordance with student progress and the instructional characteristics of lessons.

A fully interactive CMIP will possess all of the above capabilities. It is conceivable, however, that the instructional program may not be conducive to or require such an array of capabilities.

Computer-Assisted Instruction Program(s) (CAIP). As with the CMIP, a wide variety of capabilities is possible for incorporation into a Computer-Assisted Instruction Program. At least the following should be available:

1. Lessons should be authored and edited on-line with a versatile, yet easy-to-use authoring language. Within the confines of this authoring environment it should be possible to separate course "data" from course "logic" for economy of authoring large amounts of lesson material, and it should be possible to define both learner and program controlled strategies, including simulation and modeling activities.

2. Actual instruction will be administered on-line by a system of remote student operated terminals.

3. Student performance data will be collected and compiled for purposes of assessing individual progress and evaluating instructional materials.

4. The program will be capable of delivering highly adaptive instruction with extensive individualized diagnosis and prescription and a high degree of learner control supported by ready student access to critical performance and curriculum parameters and advice functions to support strategy building.

5. A variety of report formats will be easily accessible for both individual and group performance.

6. Student records will be maintained on-line and updated in real time.

An impressive configuration of media could be linked to the basic CAI hardware, but would be limited by cost considerations and media requirements. In addition, several different student response modes could be made available (light pens, touch sensors, keyboards).

It would be possible to provide for complex simulation and gaming exercises through a CAIP. These could either be designed for individual simulation environments or for those allowing interaction of several students over the system, with structure provided by the program as needed.

Programs Peripheral to Mainstream ISD Activities

Historical Trainer Data Program (HTDP). The Historical Trainer Data Program assists users in collecting and organizing pertinent data concerning previously used trainers so comparisons with current needs can be made.

The HTDP should cue and advise the user in the collection of the necessary data, such as

1. Characteristics of available trainer devices.

2. Demonstrated effectiveness of devices in training of specific objective types.

3. Cost, maintenance, and repair considerations.

The HTDP should structure the data input and automatically provide some predefined comparisons between devices. The program should also allow for specification of unique comparisons and for "what if...?" manipulations.

The HTDP should output the predefined and unique comparisons in some easily understood format. The program should also help format and generate (automatically, whenever possible) the formatted report and its updates relating to historical trainer data.

The HTDP must allow on-line review/comment and continual updates. The data input and output by the HTDP should be referenced for use in later programs such as the Trainer Requirements Determination (TRD) and the Procurement of Trainers Program (PTP).

Trainer Requirements Program (TRP). The Trainer Requirements Program will guide the user through a process of collecting and analyzing the data needed to determine the types of trainer devices best suited to the training requirements of the specific program being developed.

The program will provide automatic input from previously input data bases and/or cue the user in collecting such data as:

1. Types of objectives identified as "device" session objectives (automatic input).
2. Hardware/software system capability requirements of each "device" session objective [according to the objective, what hardware or software does the student need to manipulate, (i.e., radar system, INCOS tray)].
3. Hands-on objectives time requirements.
4. Proportion of hands-on training time to total syllabus length.
5. Instructional presentation requirements for each trainer hardware/software systems requirement category (i.e., display requirements such as audio, line-drawing, visual motion, etc; student response requirements, such as written or verbal, point, manipulate hardware, etc.; response detection and evaluation requirements, feedback requirements).

Other data is collected about available or previously used trainer devices, such as:

1. Sensory output capabilities of the device (i.e., how does it compare to hardware/software requirements of objectives, motion simulation, amount of motion, etc.).
2. Capabilities of the device to meet instructional presentation requirements.
3. Student evaluation data on use of each device in previous training settings. (Did students master similar objectives on the device?)
4. Comparative cost and maintenance data.

Other budget and time restriction data is gathered about the project itself. All of the collected data must be compared and analyzed in order to select the combination of devices which will best meet the instructional objective requirements as well as constraints on time, budget, implementation, etc. The goal is to use the least sophisticated, least expensive device suiting the purpose.

Once such data is collected, the program will perform analyses, sorts, and data consolidations, and structure the data in such ways as to facilitate the necessary comparisons. As the analysis involves numerous variables, it is imperative that the program allow "what if...?" manipulations so a range of possible solutions can be explored, if needed.

The program needs to have on-line review/comment and update capabilities. Its final output is a structured report which makes recommendations as to the mix of devices best suited to the project and the number of each such device required. This report also estimates costs and sets down specific requirements for each device which can be used by a manufacturer potentially producing such a device. The program should assist the user in developing this report with the basic text editing and manipulation capabilities of the system.

The system should cross-reference this program with other programs so each objective may be identified by the selected trainer device.

Procurement of Trainers Program (PTP). The Procurement of Trainers Program assists project planners and managers with contracting for the development of new trainer equipment or for the modification of existing trainers.

The program should automatically supply pertinent data from previously input programs (HTDP, TRP) and cue and advise users in collection of any additional data. The program should assist users in defining the final specifications for trainer devices and procurement packages. The program should allow for comparison of incoming proposals and provide a scoring system for proposal evaluation. "What if...?" manipulations should be available as part of the evaluation process.

The program needs on-line review/comment and update capabilities. Its final output is the Trainer Procurement Package which will be sent out for bids. This should be supported by the system's text manipulation and editing capabilities.

Tactics Package Program (TPP). The Tactics Package Program provides for structure and organization of the tactics package.

If the tactics package has been previously developed or is being developed by another group, the TPP should permit users to easily locate those parts of the tactics package which are pertinent to the instruction presently under development. The program should catalog or index the entire tactics package. If possible, the data base should also contain synopses of material in the tactics package.

If the tactics package is being developed in conjunction with the instructional development, the TPP should provide a historical data base to assist users in creating the tactics package for the weapon system in question. It would then provide for on-line input, review/comment, and editing of the tactics package. It would cross-reference data in the tactics package, with tasks listed in the Task Listing and objectives appearing in the Objectives Hierarchies. Thus any change made in the tactics package could be reflected in the instructional materials and vice versa. In this way, conflicts between procedures taught in training and procedures appearing in the tactics package, as well as duplication of efforts, could be avoided. The program could also assist in producing the hardcopy version of the tactics package.

Master Plan Program (MPP). The Master Plan Program will assist the user in developing a long-range management plan for the particular ISD project to be undertaken.

The MPP will automatically receive critical data input from the Problem Analysis Program (PAP). It will cue the user in the collection of additional categories of information (such as definitions of activities to be performed, resource requirements for each activity, criticality of each activity regarding completion of succeeding activities, etc.) and supply advice as to possible sources of such information. The program will organize and structure the data input so as to best facilitate the needed comparisons and analyses. The program will need on-line review/comment and edit capabilities to allow for periodic update of the master plan to reflect the most current information.

Outputs of the MPP include project time line, a list of project assignments (personnel requirements), and a project PERT chart. The program will assist the user in the on-line development of these outputs and must allow considerable

"What if...?" manipulations. The "What if...?" manipulations will allow the project planners to consider all possible variations in allocation of resources, to explore the effects of adding or losing personnel, shortening or lengthening the time line, adding new goals to the project or leaving out some possible goals, and to predict consequences of these actions to management plans.

Another output of the Master Plan Program is a final report which formally presents this plan. The MPP will assist in the structuring, development, editing, and printing of this document and its updates by using the text editing and manipulation capabilities of the system. Previously generated reports should be available as references.

Procurement Package Program (PPP). The Procurement Package Program will assist the project planners in determining what portion of the proposed ISD activity needs to be contracted to outside groups. It will also assist in securing the necessary proposals for such goods and services and in evaluating the incoming proposals.

The program will cue the users in collecting the data (some of which can be automatically supplied from earlier programs) necessary for 1) evaluating internal capabilities for successful completion of given aspects of the ISD effort and 2) determining those goods and services which need to be contracted. A job aid in the form of a checklist would guide the user in collecting such data, securing and filling out the proper forms, and developing clear, precise, and complete Requests for Proposals (RFPs) which define needed goods and services.

The PPP will assist the users in establishing a scoring system for proposals received and aid them in making the necessary comparisons and evaluations of proposals. The program will need to allow for defined "What if..." manipulations to compare complex options, such as hiring and training new staff internally, with contracting for such services from outside, or evaluating various combinations of proposals received.

The program will also offer support in the development and production of all documents (such as RFP's and reports stating final recommendations) needed for the Procurement Package. The structured report capability of the system can be used to automatically generate the large portions of such documents which are prespecified and standardized.

Navy Training Plan Program (NTPP). The Navy Training Plan Program assists project planners in collecting the necessary data and in producing the required Navy Training Plan (NTP) reports.

The NTPP will automatically supply pertinent data from previously input programs (such as the Training Program Master Plan) and will cue users in collection of additional information (such as Navy sites where the training activities may occur and what facilities are needed; Navy personnel to be involved and their relationship to outside contractors; time and place of lesson implementation). They system will give advice as to possible sources of such information. References to help locate such sources and previously developed Navy Training Plans should be available.

The program will need "What if..." manipulation capabilities for easy examination and comparison of all possible options for resource allocation. On-line review/comment and edit/update capabilities are also required.

The program will also assist the user in writing and formatting any required formal documents. Standard text editing and manipulation capabilities will therefore be needed.

Progress Monitoring Program (PMP). The Progress Monitoring Program is principally designed to track and report time spent by personnel working on each task in the ISD process. It also provides for projecting task completion dates based on availability and distribution of personnel with input from the Master Plan Program (MPP) data base.

Provisions will be made for a machine-readable timesheet, to be filled out by all personnel, which indicates time spent on each task performed. Tasks will be coded in accordance with the TPMP for ease of tracking and cross-referencing. Data from these sheets will be automatically read into and compiled in the Project Management (PM) data base.

It will be possible to input actual and anticipated availability of personnel into the PM data base. This will reflect such things as leave requests, transfers, and illness or other unexpected absences. In this way all personnel movement can be reflected in reports and time estimates, and the PPMP can serve a necessary personnel accounting function.

The user will be able to use the program to assess the impact of hypothetical personnel assignments on the completion of tasks or groups of tasks specified in the MPP. This will be done through the "What if...?" capability.

The program will be capable of providing hard copy report printouts which indicate tasks that are behind, on, or ahead of schedule as delineated in the MPP. The program will also be able to detect potential scheduling problems in advance based on anticipated resource availability compared with task requirements. In this way the user will be able to constantly monitor the pulse of the total ISD effort.

Description of the Data Bases

The user will interact with the CATSDM system using the programs outlined in the previous section. However, once data has been entered into the system, it will be organized, stored, and accessed in relationship to a number of data bases. The purpose of this section is to present a series of tables outlining the contents of each of the data bases. For each piece of data in each data base, the program used to enter that data will be identified, and the programs which access the data will be listed.

Analysis Phase Data Bases

PADB. The data base for the Problem Analysis Program consists of the following four major data items. Statement of program goal(s), operational/development constraints (e.g., time/money estimates), available or required resources, and student characteristics. These items are input by the PAP. The PAP data base must also contain built-in information on needed data and possible sources of data. The contents of this data base are used as inputs to many subsequent programs/data bases.

TLDB. The data items in the Task Listing Data Base include statements of competency or responsibility areas, specific task and component subtask requirements, and task interrelationships. Some of this information may be obtained from the PADP. This information is used in the TVSP and OHP. The TLDB must also contain information necessary to explain the task listing development process and any terms involved.

SELDB. The data items stored in the Student Entry-Level Data Base constitute a profile of the student group and include preskill measures, aptitude scores, previous job duties, ratings and/or pay scale. Much of this data will be in quantitative form. Some of the data on student characteristics, which is part of the PADB, may be incorporated into the SELDB. The SELDB is accessed by TVSP.

TVSDB. The data base for the Task Validation and Selection Data Base will include the various tasks provided by the TLDB which have been ranked according to various training categories. In other words, the data in the TVSDB is the same as the TLDB except that it has been validated and sorted. Furthermore, data from the SELDB may be used to further categorize the TVSP data. The information in the TVSDB is used by the OHP and MSP.

OHDB. The Objectives Hierarchy Data Base consists of a set of objectives for each of the tasks stored in the TVSDB and their hierarchial structure. This data is used in the MSP, SDP, TSRP, and LSP.

EMEDB. The Existing Materials Evaluation Data Base consists of information indicating sources of existing materials/equipment related to the objectives in the OHDB. It must also contain information which can suggest potential sources of material for the type of objectives in the OHDB.

Design Phase Data Bases

MSDB. The Media Selection Data Base includes data indicating the media attributes related to each objective in the OHDB and the media alternatives derived from these attributes. This data would most likely be stored in table formats. The MSDB is used in a number of subsequent data bases.

SDDB. The content of the Syllabus Development Data Base involves detailed outlines of the lesson and segment levels and the lesson interrelationships (most likely in the form of course maps). The SDDB also includes class schedules and student timetables. The SDDB utilizes data from the OHDB, MSDB, TRDB, and is accessed by the TSRP and LSP.

TSRDB. The Training Support Requirements Data Base includes a complete range of resource requirement tables indicating quantitative estimates of personnel, equipment, facilities, or services required for the development of courses. It also should include specifications for evaluation, revision and updating of materials. This data can be obtained from previous data bases (PADB, OHDB, MSDB, SDDB) with some elaboration/clarification by the TSRP input. The TSRDB is used by the ISDP.

LSDB. The Lesson Specification Data Base consists of lesson design guides which include objectives, help, examples, practice items, test items, special teaching strategies, and descriptions of graphics required. The LSDB also includes status reports which indicate the status of the various components of lessons under development. The LSDB utilizes data from the SDDB, MSDB, OHDB, and TPDB. It is accessed by the ISDPDB.

IPDB. The Implementation Plan Data Base consists of lists of training/instructional procedures, evaluation methods, resource allocation information, and scheduling data. Most of this data is drawn from other data bases (e.g., LSDB, TSRDB). This data base is used by the QCP.

QCDB. The Quality Control Data Base is composed of specifications for the collection and analysis of evaluation data as well as the records already collected. These specifications can be abstracted to constitute an evaluation manual or plan. The QCDB is used in the LTP.

Development Phase Data Bases

LADB. The contents of the Lesson Authoring Data Base is the actual lesson text along with any revision information received from lesson tryouts, and instruction for graphics, trainer use, etc. The LADB is used by the LPP.

LPDB. The Lesson Production Data Base consists of the formatted lessons and scripts for audiovisual media. Also, each lesson will indicate its current production status.

LTDB. The contents of the Lesson Tryout Data Base consists of two major items; a detailed step-by-step plan for trying out the produced lesson materials and complete records of reviewers' comments and student performance/attitudinal data for each lesson tested. The LTDB will also contain revision specifications. This data will contribute to the QCDB.

Implementation Phase Data Bases

CMIDB. A Computer-Managed Instruction Data Base would include test items, student performance data, learning prescriptions, and personnel/physical resources. The CMIDB would be accessed by the SELP and the QCP.

CAIDB. A Computer-Assisted Instruction Data Base would contain lesson text and student performance data. The CAIDB would be cross-referenced to the MSDB, OCDB, and LPDB.

Data Bases Peripheral to Mainstream ISD Activities

HTDDB. The Historical Trainer Data Program Data Base contains the characteristics of other available trainers, effectiveness in achieving specific categories of instructional objectives, and cost/maintenance data. Much of the content would be quantitative data. Some of the data could come from the NTPDB. The HTDDB is used in the TRP.

TRDB. The data collected by the Trainer Requirements Data Base consists of detailed information on the trainer requirements for the tasks or objectives stored in the TLDB or OHDB. This would include sensory limitations of devices, previously evaluated usefulness of trainers, and estimated times for certain trainer exercise. This data is merged with data already present in the HTDDB, to produce a complete data base for each trainer. This data base is cross-referenced to the NTPDB and SELDB and accessed by the TSRP, SDP and EMEP.

PTDB. The Procurement of Trainers Data Base consists of the final specifications for trainer devices and for trainer proposals. This data is based upon information from the HTDDB and TRDB.

TPDB. The Tactics Package Data Base includes information specific to the operation/training of a particular weapons system. This data is cross-referenced to the relevant tasks in the TLDB and objectives in the OHDB.

MPDB. The Master Plan Data Base consists of definitions of task activities, resources needed for each activity, sequencing of activities, and activity interdependence. Much of this data can be extracted from the PADB. The MPDB is accessed by the PPP, TLP, and QLP.

PPDB. The Procurement Package Data Base consists of statements of goods/services to be furnished (in RFQ form) along with a scoring key for completed proposals. The basis for this can come from the TSRDB and MPDB.

NTPDB. The Navy Training Plan Data Base contains information on existing Navy personnel, sites, equipment, and materials which can be used in the present development effort. The NTPDB is accessed by the TLP, EMEP, SDP, HTDP, and TPP.

PMDB. The Progress Monitoring Data Base contains actual time spent on completing tasks as well as estimated task completion data. Also contained in the PMDB will be information on personnel leaves or transfers. Some of the data in the PMDB can be obtained from the MPDB.

TABLE 1. SUMMARY OF CATSDM DATA BASES

DATA BASE	CONTENTS	ACCESSES	ACCESSED BY
PADB	program goals, constraints, resources, data sources		TLP, TLVTSP, OHP PPP, SELP, TSRP, QCP
TLDB	competency analysis, task/subtask requirements, task interrelationships	PADB	TLVS, OHP
SELDB	student profiles, pretest scores, aptitude scores, job related data	PADB	TLVS
TVSDB	task survey data, validated task listing	TLDB, SELDB	OHP, MSP
OHDB	task/subtask objectives, objectives hierarchy	TVSDB	MSP, SDP, TSRP, LSP
EMEDB	sources of materials, equipment, services	TVSDB, OHDB, PADB	DTPPB, PP
MSDB	task media attributes, task media alternatives	OHDB, TVSDB	EMEP, LSP, SDP, TSRP
SDDB	outlines of lessons and segments, lesson interrelationships, class/student scheduled	OHDB, MSDB, TRDB	TSRP, LSP
TSRDB	resources requirement tasks (personnel, equipment, facilities, specifications for evaluation revision and updating	PADB, OHDB MSDB, TRPDB	ISDAP
LSDB	lesson design guides, lesson development status	SDDB, MSDB, OHDB, TRDB	TSRP, ISDP
IPDB	institutional procedures, evaluation methods, resource management and scheduling data	TSRDB, LSDB	QCP
QCDB	specifications for data collection, analysis; evaluation plan, records	TPDB, MPDB	LTP
LADB	lesson text, lesson version data, media specifications	SDDB, TPDB	LPP
LPDB	lesson formats, scripts, production status	LADB, SDDB	LTP
LTDB	lesson tryout plans, lesson tryout feedback (comments, suggested revisions), student performance/attitude data	QCDB	LTP
CMIDB	test items, student performance data, learning prescription, learning resources	SELDB, EMEDB, TSRDB, MSDB, IPDB	QCP, SELP
CAIDB	lesson text, student response data	LPDB, MSDB	QCP, EMEP
HTDDB	trainer characteristics, effectiveness, cost, maintenance data	DTPPDB	TRP
TRDB	detailed trainer requirements, student capabilities	SELDB, HTDDB DB NTPDB	TSRP, PTP
PTDB	specifications for trainers	TRPDB	TSRP
TPDB	system-specific, operational data		OHP, TLP, HTDP
MPDB	task activity definitions, resources needed for activities, sequencing of activities	PADB NTPDB	TLP, PPP, QLP
PPDB	statements of goods, services to be furnished, proposal scoring key	TSRAPDB	PMP
NTPDB	existing Navy resources		EMEP, TLP, HTDP, HTDP, TRDB, TSRP
PMDB	resources usage, (time, money, facilities), and projections of personnel availability	MPDB	TSRP

SECTION III

ANALYSIS OF VTS IN LIGHT OF CATSDM RECOMMENDATIONS

OVERVIEW

These reports were generated from the FRAMP and FRS Aircrew Users' Manuals, Subsystem Specs for Aircrew and Enlisted Men, VTS Functional Description and CATSDM specs.

Questions were cleared up through discussions with code 3143 who were very generous in their comments and evaluation.

Due to the fact that the VTS is a dynamic system, that is, one to which additions and revisions continue to be made, only general comparisons can be made at this point. These have been refined as time and resources allowed.

There are presently different versions of the system available to different portions of the service with no one site having all the enhancements available. This report will consider the superset VTS containing all known capabilities.

COMPARISON OF STANDARD PROGRAM CAPABILITIES

Description of Program Capabilities

These standard capabilities, when available, are available mainly for areas which the system covers.

The system was designed as a PQS and CMI system with extensive administrative support and resource scheduling roles, and covers most areas of personnel management and resource management of training.

Interactive Protocols (Menus, Prompts, Aids, Advice).

The system is designed with the user in mind. There are a variety of menus available and manipulation of the system should be easy with appropriate documentation and a minimum of experience.

Cues for data entry and handling, and user assistance appear to be adequate.

Advice.

VTS does have available an advice function written to be accessed on the terminal. Information appropriate to the user's location in the system will help the uninitiated.

Library.

A library of all manuals and training materials may be implemented on the system but more in the form of an inventory than a library. Since the basic capability is there the addition of cross-references may cause a large data entry problem but no programming obstacles.

Error Detection/Reduction Routines.

Appropriate error checking of data is covered. The very nature of errors precludes absolute recovery but much work has gone into error prevention.

Automatic Input of Data from Previous Inputs.

The data base management system available makes possible the exclusion of redundant data storage.

Since VTS does not have Instructional Systems Design nor Computer Assisted Instruction capabilities at this time, there is no previous data on the system from which to draw and some of the need for that data is reduced.

With the current system capabilities and the addition of ISD/CAI capabilities the transfer of data will be possible with a little additional programming effort. Actually the data transfer procedures will probably be done as an inherent part of the ISD and CAI capabilities.

Structured Input.

Data entry and update do take advantage of structured input capabilities. With specific (ADDS 200) terminal hardware this capability is enhanced. With some additional programming this enhanced capability may be given to some of the less elaborate hardware.

Automatic Numbering and Referencing.

Presently this is not possible. Some programming effort would be necessary to provide this ability although with systems configured as at present, without ISD and CAI, this capability is low priority.

Hierarchy of Users.

System protection from users is rather elaborately provided for in both the areas of software and data base.

On-Line Review/Comment.

On-line review and comment, with appropriate protection, is standard on VTS. Changes must be okayed but comments may be left regarding any material a person is authoried to access.

On-Line Update.

Standard updating procedures are on line under the authority of specifically authorized people only.

"What If...?"

The ability to simulate different work student loads and system resources and project schedules is available. This concerns mainly CMI and PQS. Instructional development is not covered.

Communications Facility.

On-line communications facilities are available under the RSTS/E operating system. They appear to cover all the necessary aspects.

Structured Report Format.

Paperwork and forms necessary for system operations are provided by the system. The capabilities may be increased with the ADDS 200 terminal and with some effort the system can be made to provide this increased capability for some of the less elaborate software.

Reading Grade Level Computation.

Presently this is not provided under VTS. The addition of this would be a fairly straight forward task.

Text Editing/Curriculum File Maintenance Capability.

This is possible on VTS, however, due to some problems with protocol, it is not and cannot be presently provided. It is part of the software package provided by the hardware vendor.

Hard Copy Printout.

The Data Base Management System provided gives rather elaborate capabilities in developing reports and listings from the data base. Hard copy of any screen display is available.

Progress Tracking.

Progress tracking in the context of instructional systems design is not available. Student progress is provided, actually as a major part of the instructional management.

Standard Data Base Management Functions.

The standard options, including QUERY (a data access method), expected in a Data Base Management System are provided by the hardware vendor and in some areas enhanced.

COMPARISON OF FUNCTIONS FOR EACH PROGRAM

Course Authoring Functions

The CAI authoring language DECAL II is part of the system delivered with the hardware although it is not a major portion of VTS. It is not yet software connected with the VTS software or data base. Extensive text editing of the type needed is not yet offered on VTS. On VTS proper, the following are not really available in useful form:

Lesson Authoring
Lesson Tryout
Trainer Requirements Determination
Media Selection

Lesson specifications programs and lesson production programs are available by generalizing existing programs. One piece at a time the connection may not be hard, on the whole the programming effort may be substantial.

Functions Which Create Reports

Most of the recommendations in this area require that part or all of the data for the reports be input from existing data bases. In the following functions the data is either not available on VTS or only partially available. To implement these functions it would be necessary to write programs to guide the user in inputting the data and then use the existing software to generate the report.

Problem Analysis
Master Plan Program
Procurement Package
Naval Training Plan
Trainer Procurement Package
Implementation Plan Development

Student Entry-Level Program

VTS will accept data on students coming into the command and will help the instructor re-evaluate students with previous records, or completely evaluate any student. Appropiate data is stored to assist in planning student schedules and projecting the result on the training plans.

Task Listing Program

VTS can partly meet this recommendation. It has software which gathers and stores a list of the major tasks and subtasks required in any billet. Since the VTS task listing is used only for billet selection, it may not be as detailed as desired for courseware development. Modification of the existing programs would probably be sufficient.

Task Validation and Selection Program

Survey evaluation programs now exist for evaluation of surveys with up to 5 options per question.

In addition, surveys may be generated with perhaps a slight modification of existing report generation and text manipulation capabilities. Since true word processing is presently not provided, some aspects may be more difficult.

Objectives Hierarchy Development Program

VTS does allow and encourage hierarchical ordering of objectives on the system. It does allow entering, deleting and updating of objectives.

Syllabus Development Program

All the software for constructing class schedules under various time and resource constraints is present on VTS. The mechanism for automatically setting up course syllabi is also available using the objective hierarchy.

Tactics Package Program

VTS at the present time does not include tactics infomation. Programs with perhaps some considerable effect would be necessary. Furthermore, considerable data collecton would be necessary for cross referencing.

Training Support Requirements Program

VTS can and does cover the domain of this program. The resource configuration and scheduling software schedules and monitor all resources required for a course. In addition, predictions are made regarding resource requirements.

Existing Materials Evaluation Program

Existing materials can be located. Evaluation and documentation of all possible materials applications is not done.

With the addition of some data collection and cross referencing programs this may be approached with a large degree of accuracy. The programming effort would be minimal but data collection would be of some concern.

The software for choosing from a variety of training materials, once the effectiveness and existence of those materials is established, is now in use.

Historical Trainer Data Program

VTS maintains an inventory of trainer equipment, status, and effectiveness. Information on the characteristics in the detail needed is not available. In order to perform the desired analysis the data base would have to be expanded and software for evaluation of characteristics in light of training requirements would be necessary.

Progress Monitoring Program

As part of instructional system design this does not exist on VTS. There are programs in existence which can be genralized to produce most of the needed output. The level of effort to produce the project monitoring required should be acceptable. Specifications and an operational prototype for such a system (AMS) have been developed within DOD.

Computer-Managed Instruction Program

In the area of CMI the following is characteristic:

1. VTS does offer a full range of on-line tests and evaluation or entry of off-line tests.

2. Student performance is saved and used for student feedback diagnosis.

3. Student prescriptions for study are given although human intervention is required.

4. Student records are managed in real time.

5. Banks of test items are stored to produce random objectives based tests.

6. Instructional materials are evaluated in light of student performance.

7. Personnel and physical resources are scheduled automatically.

Computer-Assisted Instruction Program

With DECAL II from the hardware vendor some aspects of CAI do become available. It is, however, a substantial effort to implement as is and as is it doesn't conform to CATSDM recommendations. Since it is not generally thought of as being included with VTS we will consider VTS without DECAL II. In this light the following becomes apparent:

1. There is no lesson authoring capability on VTS.

2. Actual Instruction is not administered on-line.

3. Some Student test performance data is collected.

4. There are a variety of formats for accessing individual and group performance data.

5. Student records are maintained on-line but not in the detail required by CAI. They are updated in real time.

   VTS does plan to have a CAI component and may at some future time fulfill all of these recommendations, however, DECAL II is seen as inadequate, as is.

Quality Control Plan

VTS does not assist in developing a variety of quality control plans. It does however, provide for a given plan and the application of various plans which may be developed off-line.

COMPARISON OF DATA BASE CONTENTS AND STRUCTURE

Description of the Data Base

The following data bases are not presently part of the VTS data base and would need new programs to be written in order to add them. Most of the above fall into two categories, one (#1), development of curriculum, and two (#2) forms generation with the goal of curriculum development.

Program	Data Base	Use Categories
Problem Analysis	PADB	#1, #2
Master Plan	MPDB	#1, #2
Procurement Package	PPDB	#2
Navy Training Plan	NTPDB	#2
Lesson Interrelationships Matrix	LIMPDB	#1
Tactics Package	TPPDB	#2
Training Support Requirements	TSRDB	#1
Historical Trainer Data	HTRDPDB	#1
Trainer Requirements	TRPDB	#1, #2
Trainer Procurement Package	TPPPDB	#2
Implementation Plan	IPDBP	#1, #2
Quality Control Plan	QCDB	#1, #2
Project Management	PMPDB	#1

In the cases where forms are developed (#2), text editing/curriculum file maintenance capabilities would be almost a necessity for computer generation and updating of these forms.

The remaining portions of the data base exist at least in part.

Student Entry Level Data Base

This data does exist and can be updated when the student arrives after being initially entered on notification of transfer. Moreover, the data is updated automatically as students progress.

Task Listing Data Base

There is a list of tasks and subtasks contained within a course hierarchy.

Objectives Hierarchy Data Base

There is a hierarchy of objectives within the course hierarchy.

Course Syllabi Lesson Level Data Base

There is a syllabus for each of four out of five levels of tasks and subtasks within a hierarchy.

Course Syllabi Segment Level Data Base

This is also available.

Existing Materials Data Base

There is a list of all existing training materials at a particular site, but this is not connected with materials at other sites and it is referenced at the course level, not the objective or task level.

What presently exists is probably adequate.

Lesson Validation Data Base

A history is kept of test questions and instructors in order to validate their effectiveness. Student histories are further kept for future digesting of this material. However, for a rigorous validation of the lesson materials more detailed information is needed and the programs would need revision accordingly.

Computer-Managed Instruction Data Base

This data base is completely there as the VTS system has been enhanced specifically for CMI. Enhancements in scope and sophistication are needed.

Computer-Assisted Instruction Data Base

Lesson material and instruction are not on-line, but student performance is kept and may be updated on-line. Group and student reports are also generated.

This addition would require a major effort in programming and planning, if it were attempted to add this to VTS. A better strategy would be to design interfaces with other systems dedicated to true CAI.

Lesson Specifications and Text Data Base

Many lesson specification elements are already part of the data base, however, more would be necessary.

SECTION IV

PROGRAM DEVELOPMENT AND IMPLEMENTATION PLAN

It is possible to generate many reasonable alternative plans for developing and implementing a CATSDM system, each with its own particular strengths and weaknesses.

If it were difficult to extract clear recommendations from the findings of this study, there might be some merit to providing some alternative plans. However, as the recommendations of this study are strong and unambiguous, only the recommended plan is outlined, with supporting rationale.

PROGRAM PRIORITIES IN LIGHT OF THE COST/BENEFITS ANALYSIS

Benefits of Automated Support to FAISD Tasks

Many factors affect the cost to benefit relationships which imply the priorities which should be afforded to various identifiable components of the CATSDM system.

These are discussed in Data Item 004, ISD Task Priorities for CATSDM, of this contract, in the "Implications" section following each of the questions in Section I of that document. In the interests of generating a "stand alone" document as this final report, these are essentially reproduced here to provide the necessary background rationale for the task selection and prioritizing efforts which follow. These may be considered to be benefits of automating the ISD process.

-Tasks which can be turned over to personnel with little or no specialized knowledge or skills, frequently will be benefited in terms of consistency and overall quality control when closely monitored or supervised. Close monitoring and specific guidance and structuring of subtasks can be provided with some effectiveness through automated management support of the task.

-Tasks which require the specialized input of a very few key people are good candidates for computer support. Extensive text manipulation support provides a production environment in which the written output of these personnel can be greatly increased and enhanced because of the much more economical editing and rewrite capabilities. In addition, the expertise of these personnel can, in some cases, provide a leverage effect. The impact of their input can be multiplied through utilizing it to design high-payoff automation functions related to the task. These can lead, in the long run, to increases in the effectiveness of their input.

-Anytime the ISD activity is involved in many simultaneous tasks, management complexity is greatly increased. The advantage of computer support in tracking the progress of the ongoing tasks and coordinating the personnel and resources they require, is obvious.

-One of the greatest deficiencies of ISD activities, both within and without Department of Defense, is the almost universal lack of provision for collecting and analyzing the sort of detailed data which would provide a good planning base for efficient use of personnel and resources and to support the extensive scheduling needs of the overall ISD activity. Computer support can provide the means for collecting the historical data base and analyzing its implications. This will allow much better plananing and scheduling of critical resources such as, in this case, key personnel who cannot easily be replaced on the task.

-The requirement for extensive training of personnel prior to the completion of the task decreases the degrees of scheduling freedom available to the management of the ISD activity since the training time must also be scheduled prior to beginning the task. In general, as scheduling problems become more complex, the potential benefits for automated system support increased.

-To the extent personnel must perform their jobs without close supervision, the ISD effort may potentially benefit from the collection of more detailed data on performance of intermediate ISD subtasks and from the capability to provide the affected personnel with more structured job aids, within-task checkpoints, and feedback. This effort can be greatly facilitated through the provision of easy access to and the requirement for, frequent interaction with the automated management system by the affected personnel.

-To the extent that a task can be broken up into easily identifiable subtasks, computer support of management makes it increasingly possible to closely monitor the progress of individual personnel within the task. This provides a data base which can guide management towards necessary restructing of the task and/or toward additional training needs in regard to specific personnel.

-To the extent to which progress checkpoints can easily and objectively be determined, it is appropriate to monitor and report on overall task progress through computer support. The benefits in terms of management and control are obvious.

-External monitoring of each task becomes more important in cases where the levels of personnel assigned to the task are unable to adequately monitor their own progress. Automation provides potential benefits in terms of information and process support to personnel.

-The importance of a real time management system is greatly increased on those tasks where the timely completion of the task is critical to on-time start-up and completion of other ISD tasks. Computer support of the management function and/or implementation of the task offers correspondingly greater potential benefits in these cases, especially in terms of early identification of problems, and in terms of exploring the effects of alternative "solutions."

-Computer support can greatly alleviate communications problems which may arise between personnel from varied specialized backgrounds by providing a central clearinghouse and definition point for all terminology used, and by providing easy access to both formatted reports and detailed raw data. The computer system can greatly streamline the information transfer process while providing a valuable discipline and consistency support function.

-If the characteristics of any task are such that progress could be significantly hindered by varied nonstandard terminology, procedures, techniques, and so on, then the potential benefits inherent in centralized data and reporting system become much more valuable.

-There may be many efficiencies effected in dividing tasks into pieces which could be worked on independently by many personnel. However, the cost of this efficient use of manpower is often management complexity. Automated systems support enables you to take advantage of your full personnel pool much more efficiently.

-On long tasks, frequent detailed reporting may have a strong motivational effect. The primary advantage is that personnel with access to frequent detailed progress reports, are able to maintain their sense of progress. Without this capability, some tasks seem to drag on forever at considerable cost to productivity and morale.

-Similarly, frequent reports may be necessary to maintain a sense of getting somewhere in those cases where the task may not be clearly related to the final project products for the personnel engaged in the task. With automated support of management, it is possible to maintain a sense of "getting somewhere."

-Tasks involving extremely well-defined procedures are very good candidates for automated monitoring and reporting and/or for real time cueing and prompting in terms of interactive job aids. This can result in better audit control (through standardized approaches and error detection) and better efficiency (through prompting and pacing).

-The more mechanical or algorithmically defined a task is, the more that task becomes a candidate for complete automation, thereby freeing staff for other activities.

-If there is a complex mix of production and training facilities required to carry out a task, there is increased probability tht the management of the task would be benefited through use of asset/facilities scheduling and management program support.

-In any task requiring a very large typing load, there is a high possibility that on-line text editing and manipulation support would be cost effective and would result in improved product quality.

-The generation of extensive reports and/or instructional products implies a need for scheduling of production assets and facilities for possibly extensive text manipulation and/or headlining, typesetting capabilities, and for the creation and maintenance of data base to be manipulated for report generation purposes.

-Extensive requirements for training facilities can generate complex scheduling problems. In the case of hands-on training devices, there will be a need to schedule students serially without conflict to take maximum advantage of the usually limited supply of hands-on devices. In the case of group classroom requirements, there is a need to schedule students in such a way that students with the proper prerequisites are scheduled simultaneously into the classroom resource. Scheduling support provides benefits through more efficient use and maintenance of expensive resources.

-Information to be collected by project personnel can be recorded and manipulated by the automated management system. In those cases where information is to be generated by the project personnel, computer support can help with cueing and prompting programs and interactive data input programs in insuring consistency of format structure and terminology.

-Properly designed automated data collection techniques tend to enhance reliability and completeness of information generated by people with wide variations in experience and background.

-If data is to be used in multiple alternative forms, it is extremely important that the data base be quickly and easily manipulated. In most cases, as the data base becomes large, this is possible only with computer support.

-If data to be collected has a high proportion of simple alphanumeric codes or repeated brief text string such as key words or abbreviations, computer support can offer valuable data packing, sorting, retrieval, and error-checking support. For those longer prose records (sentence or paragraph length or longer), large-scale storage, high speed retrieval and output, and complex text editing capabilities are important advantages to be offered by computers.

-The importance of the sophisticated error-checking and correction capabilities available under automated systems support is hard to deny. Considerable time and expense can be saved as more and more errors are detected immediately when made, or as errors become easier and easier to correct.

-In cases where it would be relatively simple and inexpensive to build in data validation processes, automating this function is potentially advantageous. In those cases where data validation is somewhat more difficult, but due to the criticality of the data, still very important, a separate decision must be made on whether or not to take the extra time and effort to automate parts of this validation process.

-In cases when data could be generated as it is needed, rather than stored, often a simple data translation program or generation algorithm can replace personnel or storage files in supplying this data.

-Often, data structures possess a high degree of built-in redundancy. For example, it is possible to document the various rationales for computer support of the individual ISD tasks by storing one copy each of a set of footnote comments, and then referencing them by a single number (as was done in data item 004 of this contract.) In this case, whole paragraphs of data were coded and referenced as one- and two-digit numbers, and then were expanded into the paragraphs they represent by the system at retrieval or output time. Such data translation and packing capabilities can greatly reduce the amount of storage required for large data bases with the proper redundancy characteristics. A library of student prescriptions in the CMI data base would be a similar example.

-On data bases of any size at all, retrieval by key word techniques or retrieval through extensive cross-referencing and indirect accessing techniques, is inefficient, time-consuming, and expensive by manual methods. This is an extremely powerful area of advantage for automated systems support.

-Automated system support offers significant advantages in any case when information to be retrieved from the project data base needs to be formatted and/or manipulated for reports or other documentation.

-The more often updates or changes to the data base occur, and the more extensive these modifications are, the greater the advantage to be realized by automation of the data base.

-Extensive numerical analysis requirements on any or all parts of the data base strongly imply that computer support would be advantageous.

-The more ways information needs to be sorted and the larger the data base to be sorted, the greater the advantage which accrues to computer support.

-Links between different parts of the data base are most easily established and maintained, and most economically taken advantage of, through automated systems procedures. In particular, merging of data bases created at different times and/or for different purposes, is most easily accomplished through cross-referencing techniques which again strongly imply computer support.

These are representative benefits of automating FAISD tasks. These must be considered, however, in light of the costs.

Cost-Related Development Priorities

This section will discuss several development priority factors, those which are based on the cost analysis of the Fleet Aviation Instructional Systems Development (FAISD) tasks, those which are then subsequently based on other evolving on-line aids to ISD such as work currently being done under Army Research Institute (ARI), Defense Advanced Research Projects Agency (DARPA), and other contracts within DOD.

1. ISD task priorities, largely cost related, were generated through an analysis of FAISD tasks in terms of manpower and resources required compared to the potential benefits of automated support for each task.

This analysis was based upon three major factors, 1) the degree to which the particular FAISD task could benefit from automated support. 2) the impact of seven "criticality" factors, and 3) the "personnel leverage factor" of the task. The relative degree to which a task could benefit from automated support was determined on the basis of a set of 48 information management concerns. Each task was rated by the study group as either benefitting or not benefitting from automated support on each of the information management areas. A score was assigned (from 1 to 48) to each task reflecting how many of the potential areas of benefit from automation applied to that task.

Seven areas were judged to offer especially important potential benefits from automation, if they applied to a task. These were rated as being worth either 2 or 3 extra points each, which were added to the total for each task as applicable.

Each task was then examined to determine the number of man-hours likely to be impacted by computer support for the task. A "personnel leverage" weight was assigned to each task based on the number and type of people affected. This was considered to be very important, and as much as 15 points were assigned to some tasks on this variable. The values assigned were subjective, but represented the judgements of experienced and well-qualified ISD personnel with backgrounds in computer applications to developing instruction.

A total score was then computed for each task and the resulting priorities are shown as follows:

Priority 1

Programs	Program number referred in Figure 1
1. Standard Capabilities	*
2. Lesson Specification Program	(10)
3. Lesson Authoring Program	(13)
4. Lesson Production Program	(14)
5. Lesson Tryout Program	(15)
6. CMI Program	(16)
7. Media Selection Program	(7)
8. Progress Monitoring Program	(25)

*Number in parentheses refers to program number used in Figure 1.

Priority 2

1. Objectives Hierarchy Development Program	(5)
2. CAI Linkage Specifications Program	(17)
3. Student Entry Level Program	(3)
4. Task Listing Program	(2)
5. Task Validation and Selection Program	(4)
6. Syllabus Development Program	(8)
7. Training Support Requirements Program	(9)
8. Existing Materials Evaluation Program	(6)

Priority 3

Programs	Program number referred in Figure 1
1. Tactics Package Program	(21)
2. Trainer Requirements Program	(19)
3. Problem Analysis Program	(1)
4. Master Plan Program	(22)
5. Procurement Package Program	(23)
6. Navy Training Plan Program	(24)
7. Procurement of Trainers Program	(20)
8. Implementation Plan Program	(11)
9. Quality Control Program	(12)
10. Historical Trainer Data Program	(18)

2. Existing VTS capabilities already address many FAISD tasks, however, and therefore may somewhat revise priorities, since they are available at relatively small, if any, additional cost.

Figure 1 shows one possible representation of the set of major program systems described in Section II, which would, when linked to each other and using the integrated data bases outlined, address the FAISD tasks. Figure 2 relates those tasks to the programs.

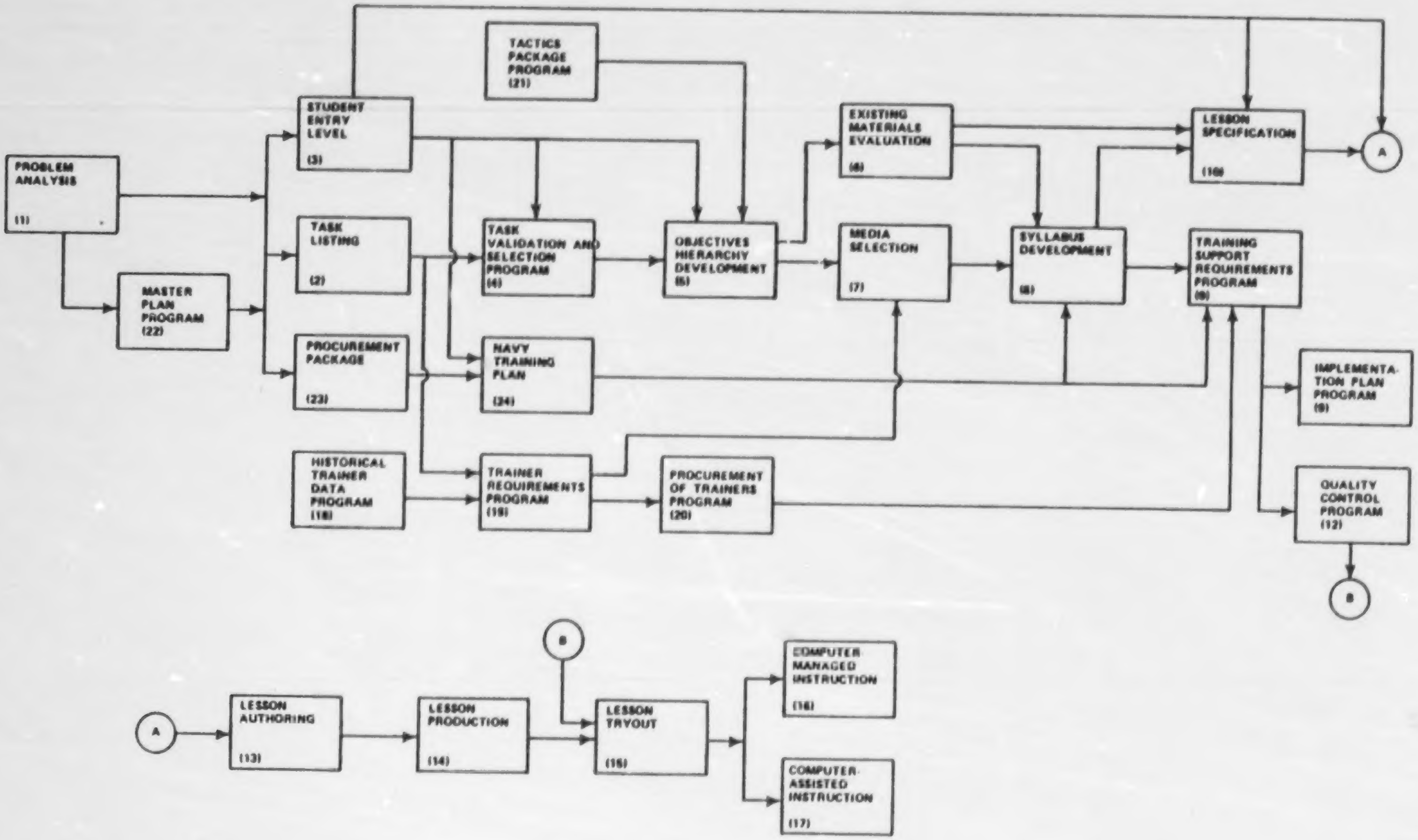
TACTICS PACKAGE PROGRAM (21)
PROBLEM ANALYSIS (1)
STUDENT ENTRY LEVEL (3)
MASTER PLAN PROGRAM (22)
TASK LISTING (2)
PROCUREMENT PACKAGE (23)
HISTORICAL TRAINER DATA PROGRAM (18)
TASK VALIDATION AND SELECTION PROGRAM (4)
NAVY TRAINING PLAN (24)
TRAINER REQUIREMENTS PROGRAM (19)
OBJECTIVES HIERARCHY DEVELOPMENT (5)
PROCUREMENT OF TRAINERS PROGRAM (20)
EXISTING MATERIALS EVALUATION (6)
MEDIA SELECTION (7)
SYLLABUS DEVELOPMENT (8)
LESSON SPECIFICATION (10)
TRAINING SUPPORT REQUIREMENTS PROGRAM (9)
A
IMPLEMENTATION PLAN PROGRAM (9)
QUALITY CONTROL PROGRAM (12)
B
A
LESSON AUTHORING (13)
LESSON PRODUCTION (14)
B
LESSON TRYOUT (15)
COMPUTER MANAGED INSTRUCTION (16)
COMPUTER-ASSISTED INSTRUCTION (17)

Figure 1: CATSDM System of Computer Programs — General FLOW Diagram (Process Monitoring Program Not Pictured Here Since It Tracks all Activities Shown)

The priority of program development suggested by the prioritized FAISD task-to-program relationships indicated in Figure 2 indicates the following preliminary program development priorities. Notice that many of the FAISD tasks would be served by the same programs (i.e. 25.1, 20.2, 19.1 on Figure 2) using different data. The analysis of VTS capabilities shows that several of the programs are either currently available in some form, or are easily adapted from existing programs, data, or basic systems capabilities with a minimum of extra effort. These include the student entry level, task listing, objectives hierarchy, syllabus development, and computer managed instruction programs. When these are taken into account, the priority listing becomes:

Priority 1

Program	Figure 2 Program Reference Number
1. Standard Capabilities	*
2. Lesson Specification Program	(10)
3. Lesson Authoring Program	(13)
4. Lesson Production Program	(14)
5. Lesson Tryout Program	(15)
6. CMI Program	(16)
7. Media Selection Program	(7)
8. Progress Monitoring Program	(25)
9. Student Entry Level Program	(3)
10. Task Listing Program	(2)
11. Objectives Hierarchy Development Program	(5)
12. Syllabus Development Program	(8)

Priority 2

1. CAI Linkage Specifications Program	(17)
2. Task Validation and Selection Program	(4)
3. Training Support Requirements Program	(9)
4. Existing Materials Evaluation Program	(6)
5. Tactics Package Program	(21)
6. Master Plan Program	(22)
7. Implementation Plan Program	(11)
8. Historical Trainer Data Program	(18)

Priority 3

Program	Figure 2 Program Reference Number
1. Trainer Requirements Program	(19)
2. Problem Analysis Program	(1)
3. Procurement Package Program	(23)
4. Navy Training Plan Program	(24)
5. Procurement of Trainers Program	(20)
6. Quality Control Program	(12)

Tasks from FAISD model*	CATSDM program(s) related to each task**
11.1 STCs Lesson Material Authored & Formative Tryout	13,13,14,16
16.4 QC Plan Implemented	15,16
10.2 STCs Lesson Specifications Development	10
8.1 Maintenance Objectives Hierarchy Development	5
10.1 STCs Generalities Development	5
7.1 OP Objectives Hierarchy Development	5
11.2 STCs Lesson Material Produced	14
25.1 Student Training at FRS	16,17
20.2 Initial FRS Instructors Take ITC (Fr)	16,17
19.1 FITT Take ITC (Tr)	16,17
23.1 Student Entry Level Determination	3
9.3 STCs Ordinal Syllabi Development	8
13.1 ITC (Tr) Developed	2,4,5,7,8,10,13,14,15
18.1 ITC(IS) Development	
20.1 Initial FRS Instructors Take ITC (Fr)	16,17
5.1 Operator Mission Analysis, Function Analysis and TI Development	2
6.1 Maintenance Task Lists Development	2
21.1 Remaining FRS Instructors Take STCs	16,17
20.4 Initial FRS Instructors Take ITC (IS)	16,17
21.2 Remaining FRS Instructors Take IT (IS)	16,17
21.3 Remaining FRS Instructors Take ITC (Tr)	16,17

Tasks from FAISD model*	CATSDM program(s) related to each task**
4.1 NTP Development	24
9.4 TSRA Development	9
2.1 Training Program Master Plan Development	22
3.1 Procurement Package (Training Section) Preparation	23
11.3 STCs Lesson Material Revised	13,14,16
12.1 Historical Trainer Related Data Collected	18
15.2 CATSDM	▨
17.1 Implementation Plan Development	11
18.2 ITC(IS) Production	14
3.3 Proposal Review and Selection	23
12.3 Procurement Package Developed for Complex Trainers	20
12.5 Other Trainer Procurement Package Developed	20
1.1 Problem Analysis	1
9.2 Media Decisions Made for STCs	7
16.1 Quality Control Plan Development	12
5.2 OP Task Selection	4
5.3 - 5.7 OP Task List Update	2
6.3 - 6.7 Maintenance Task List Update	2
7.2 - 7.5 OP Objectives Hierarchy Update	5
8.2 - 8.5 Maintenance Objectives Hierarchy Update	5
12.2 Preliminary WST & Other Trainer Requirements Determination	19

Tasks from FAISD model*	CATSDM program(s) related to each task**
9.7 - 9.9 STCs Syllabi Update	8
15.1 CATSDM Plan Development	▨
6.2 Maintenance Task Selection	4
20.3 Initial FRS Instructors Use Practice Using Trainers	16,17
9.1 Media Selection Models Development	7
9.5 TSRA Review	9
9.6 Resource Requirements List Development	9,11
13.2 ITC (Tr) Revised	13,14,16
4.3 NTP Revision	24
10.3 - 10.4 STCs LSs Update	10
12.7 All Trainers Tested at Factory	25
12.8 Trainers Installed at FRS, Debugged & Accepted	25
14.1 Weapons System Fleet Tactics Package Developed	21
14.2 Weapons System Fleet Tactics Package Update	21
2.2 - 2.8 TPMP Update	22
4.2 NTP Review Conference	24
1.2 - 1.8 Problem Analysis Update	1
17.2 - 17.3 Implementation Plan Updated	11
16.2 - 16.3 QC Plan Update	12

*Tasks are ordered in accordance with the priorities established in Deliverable No. 0004

**Program numbers correspond to those shown in Figure 1.

Figure 2: Correspondence Between Prioritized FAISD Model Tasks and CATSDM Programs

3. Other on-line aids to ISD are under development within DOD. With one major exception, these are not yet complete or available. That exception is an Author Management System (AMS) completed under DARPA Contract #MDA 903-76-C-0216. The AMS is equivalent to the Lesson Production Program, already a Priority 1 program, so the priorities do not change. However, since the completion, availability, and applicability of these other efforts could be very important to CATSDM in terms of changing priorities 2 and 3, in tasks 5.0, 5.1, 5.2, and 5.3 of the tasking and phasing chart in Figure 3. It is suggested that these efforts be monitored throughout the CATSDM development effort.

The tasking laid out in the tasking phasing chart shown represents the study group's best judgement of a reasonable time frame in which to accomplish each task shown. It is understood that the hardware upon which CATSDM is implemented, the software facilities available for CATSDM development on that system, and the level of detail, sophistication, and complexity finally decided upon for the emerging CATSDM system, all will have an impact upon the level of effort required. It is felt, however, that within reasonable limits these degrees of variability can be accommodated in terms of increasing or decreasing staff and still be accomplished within the approximate time frames shown. Notice that the tasking is set up in terms of a three year initial development program and, of course, any ongoing maintenance revision, and enhancement which would be desirable would continue beyond that period. It is the feeling of the study group that in this implementation schedule, the end of the first year would show maximum payoff, in that the highest priority functions and capabilities would then be available for use.

In this suggested implementation plan, three major areas of development have been selected for preliminary implementation. It is felt that if the standard capabilities are provided in the first phase, and if the Author Management System and CMI systems are enhanced and expanded by development of the selected Priority 1 programs suggested in this document, the major part of the CATSDM system can be provided by the end of the first phase. Again, it should be realized that the number of priority one programs attempted within this context, the level of detail and sophistication aspired to on this first cut, and other factors all will influence, at the very least, the level of manpower which must be applied to the task; and potentially would, obviously, significantly, impact the time phasing of this implementation plan. It should be realized that the implementation plan suggested assumes that all of the Priority 1 programs are to be attempted in the first phase at a level of sophistication and detail which the planning

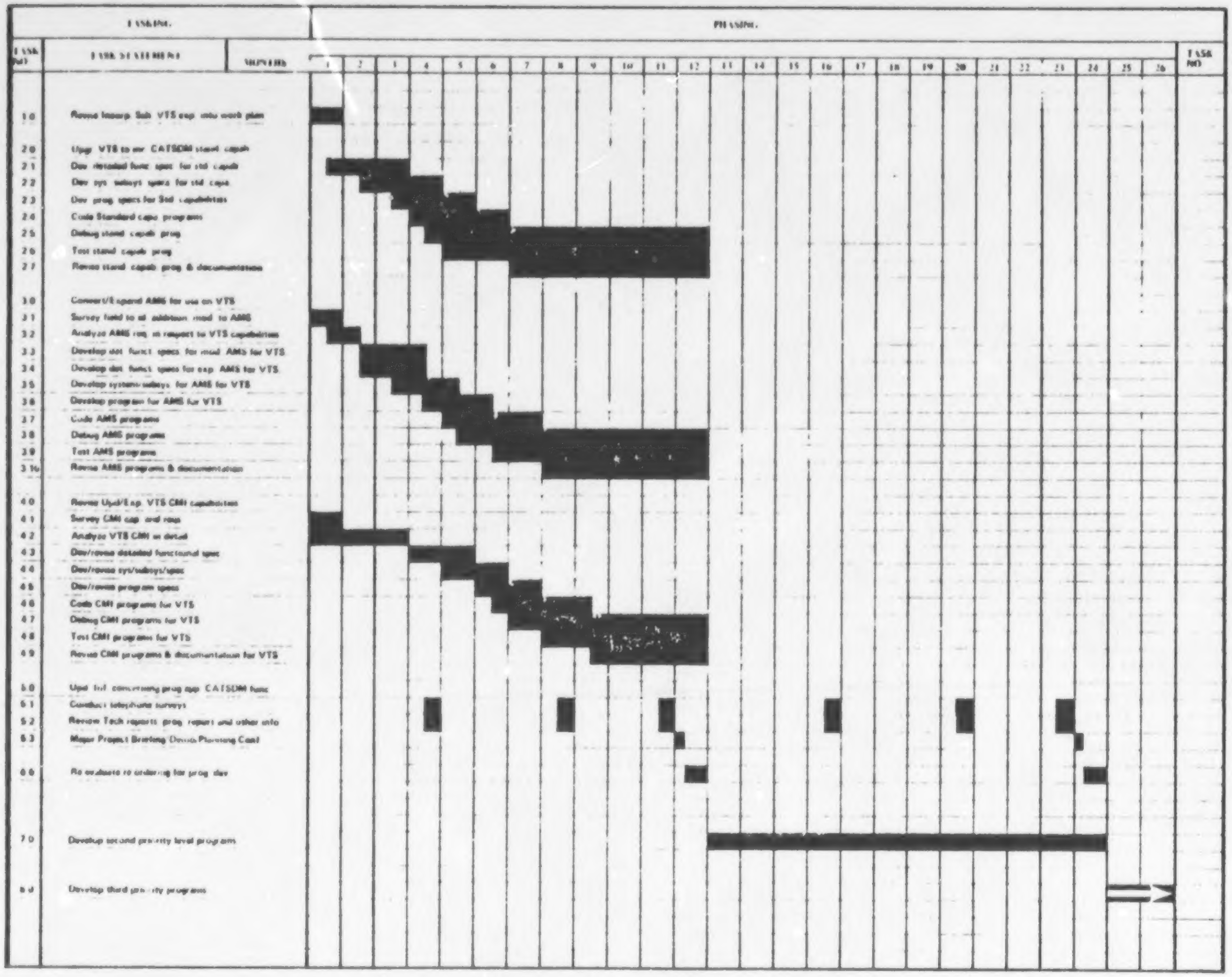

Figure 3: Phasing and Tasking Chart for CATSDM Development

group believes would provide a relatively high level of ISD support service, but perhaps without all of the "bells and whistles" that the eventual, mature, CATSDM system would exhibit. It is further felt that a program/system specification, design, and development level of effort reasonable to accomplish the Priority 1 programs in the first year, as shown on this tasking phasing chart, could reasonably be accomplished with about ten man years of effort.

In the following explanation of the tasks, it should be understood that the individual priority one programs do not cleanly fall singly into the basic capabilities AMS of CMI subsystems, in most cases, but rather they distribute across those and would primarily be considered as additions, modifications, or enhancements to those systems. It is also felt that in the case of all three areas of concern, existing capabilities provide a good head start on most of the Priority 1 programs. Otherwise, their design development and implementation, in the time frames and within the level of effort suggested, would clearly be somewhat unreasonable.

Task 1.0 Revise/incorporate scheduled VTS expansion into work plan. It is clear that in the event CATSDM is implemented upon VTS that the CATSDM development program itself will constitute a non trivial perturbation within the existing VTS expansion and development schedule. Therefore, it is felt important that an accommodation be reached as early as possible between these potentially conflicting activities. Possible outcomes of this planning activity would be a revision of this work plan for CATSDM, or a revision of the VTS expansion schedule, or both.

2.0 Upgrade VTS to incorporate CATSDM standard capabilities.

2.1 Develop detailed functional specifications for standard capabilities. In this task the general functional specifications listed for a CATSDM system would be explained in more detail and enhanced as necessary, taking into consideration any built in advantages or disadvantages or special features of the particular hardware/software system selected.

2.2 Develop systems/subsystems specs for standard capabilities. At this time it will be possible to begin developing the system/subsystem specifications for the standard capabilities in any cases where new software development or existing software modification is called for.

2.3 Develop program specifications for standard capabilities. As the systems/subsystems specifications become better, development of the program specifications can be carried out.

2.4 Code standard capability programs. In all cases where standard capabilities are not already existing on the system, but are possible to program into the system being used for CATSDM development, coding can begin as soon as the standard capability program specifications are well developed.

2.5 Debug standard capability programs. As coding of the individual programs involved is complete, individual debug of those programs can proceed. As multiple programs are finished, system debug can commence in terms of debugging the interaction of those programs with one another and the existing hardware/software system.

2.6 Test standard capability programs. When the various levels of debug are complete or well advanced, testing of the standard capabilities can commence.

2.7 Revision of standard capability programs and documentation. From the testing activity, revisions or additions may be indicated, these can then be programmed and final documentation of the standard capabilities can be completed.

3.0 Convert/expand Author Management System for use on VTS.

3.1 Survey field to identify additional useful or necessary modifications to the Author Management System. The Author Management System exists in the form of an operating prototype with extensive documentation. However, by the time it is incorproated as a major program function in the CATSDM system, it may be possible to identify, in the ISD field, desirable additions or modifications which could be incorporated at this time.

3.2 Analyze AMS requirements in respect to VTS capabilities. A critical part of accommodating an existing operational system such as the AMS onto a different hardware/software system is an analysis of the target systems' capabilities, in respect to the requirements (either implied or explicit) of the system being translated. It is felt that this can be accomplished in a fairly short time.

3.3 Develop detailed functional specifications for the modifications for AMS for VTS. In those cases where AMS is to be modified to accommodate specific requirements of VTS, detailed functional specifications will be prepared.

3.4 Develop detailed functional specifications for expanding AMS on VTS. In those cases where AMS is to be expanded as a result of the survey of the field with new functions or capabilities, detailed functional specifications will be prepared for these enhancements.

3.5 Develop system/subsystems specifications for AMS. Following the detailed functional specifications, systems/subsystems specifications for the AMS, for the modifications to AMS, and for the enhancements to AMS, will all be prepared.

3.6 Develop program specifications for AMS. When system/subsystems specifications are complete or well advanced, program specifications can begin.

3.7 Code AMS programs. When program specifications are complete or well advanced, coding of the individual AMS programs can commence.

3.8 Debug AMS programs. As coding of individual programs is completed, debug of those programs, both singularly and in conjunction with one another, can be started.

3.9 Test AMS programs and systems. As debugging of individual programs is complete, testing of the AMS as a system operating on the host hardware/software system can begin.

3.10 Revise AMS programs and documentation. On the basis of the testing results, revisions to programs may be required and documentation of the final AMS system will be completed.

4.0 Revise/update/expand VTS CMI capabilities.

4.1 Survey of CMI capabilities and requirements. A survey will be conducted of existing CMI applications, both within and without the military, specifically in regard to the needs of the potential CATSDM audience, and a report of desirable CMI characteristics discovered or identified will be prepared.

4.2 Analyze VTS CMI in detail. Simultaneously, the existing CMI capabilities on VTS will be analyzed in some detail, and in conjunction with the survey, a comparison will be made on each desirable characteristic in terms of its availability or adaptability on VTS.

4.3 Develop new or revise existing detailed functional specifications. Where additions on enhancements to the existing VTS CMI capabilities are suggested, detailed functional specifications for those additions or enhancements will be prepared and incorporated into the functional specifications for the existing CMI system.

4.4 Develop new or revise existing systems/subsystems specifications. Similarly, enhancements or revisions to the systems/subsystems specifications will be prepared as necessary.

4.5 Develop/revise program specifications. The program specifications will be either generated or revised, and integrated into the existing program specifications.

4.6 Code CMI programs for VTS. With the completion or advanced development of the program specifications, coding can begin on CMI modifications.

4.7 Debug CMI programs for VTS. The additions or revisions can be debugged as coding advances.

4.8 Test CMI programs for VTS. The revised VTS CMI program can begin test and evaluation as debugging is complete.

4.9 Revise CMI Programs and Documentation for VTS. As testing advances, programs can be revised as necessary and the documentation for the CMI subsystems can be updated appropriately.

5.0 Update information concerning progress of applicable CATSDM functions.

5.1 Conduct telephone surveys. On a periodic basis throughout the CATSDM development activity, information needs to be gathered and updated concerning other DOD efforts which could be adapted or incorporated within the CATSDM system. One part of this information gathering process will be telephone surveys to other DOD activities.

5.2 Review technical reports, progress reports and other information. Augmenting the telephone surveys will be literature reviews and review of such documentation on other activities as may be available.

5.3 Major project briefings/demonstrations/planning conferences. Towards the end of each major phase of the CATSDM development, a major project briefing should be held in which a demonstration of work completed within the phase can be given in terms of actual exercise of the functions. This should take place in the context of a planning conference in which an evaluation of the progress by the planning group would generate a revised tasking and phasing for the next major phase of the project.

6.0 Re-evaluation and reordering of priorities for program development. On the basis of the information available concerning the progress made in each phase of the project, and upon the information available concerning other DOD activities which may be applicable, a revised list of priorities for program development in the next major phase of the project will be issued.

7.0 Develop second priority level programs. On the basis of the reordered priorities, a work plan for the second phase will be developed and, as appropriate, work may begin on the development of the second priority level programs.

8.0 Develop third priority programs. Following a second phase major project briefing/demonstration/planning conference and a re-evaluation and reordering of priorities for program development, planning and work for the third phase will proceed.

DEVELOPMENT AND IMPLEMENTATION RESOURCE REQUIREMENTS

Several factors will have considerable effect on the resources required to develop and implement a CATSDM system. One is the extent to which the system base from which it is to be evolved already meets many CATSDM needs. VTS, the most attractive candidate, will already support a range of CATSDM needs. However, the VTS is itself an emerging system, and the status of documentation is such that the full extent to which the VTS meets CATSDM needs is not precisely clear.

Another factor is the availability of other CATSDM-related functions, developed within DOD, which could be adopted or adapted, as well as the extent to which existing VTS functions will need to be adapted to be used. Obviously some, such as the Author Management System (AMS) developed for DARPA, are prime candidates for early attention. Others will have to be studied before a precise estimate of resources required to make them useful could be made.

Another factor affecting needed resource levels is the extent to which CATSDM specifications could be incorporated into future development work already scheduled (such as the continuing work on VTS) without significantly increasing those costs. This, too, cannot be known at this time.

And finally, this class of problem, that of estimating costs and resources required for a major, complex, sophisticated, information management system for use by a wide range of personnel, is essentially open-ended. That is, while there may be some useful minimum threshold below which a useful system is just not possible, it is possible to add refinements, sophistication, and human engineering considerations which would in some environments be very important, and which would effectively increase costs tenfold over that threshold.

However, based on the assumptions listed under the previous section, upon the experience of this author with developing a major CATSDM subsystem under DOD funding already, and upon the high level of expertise of the VTS staff and relative utility of VTS as it is currently understood, the following estimates are offered with some confidence.

Hardware/Software Resources

The VTS, as is, is potentially a reasonable CATSDM environment with a few reservations. One is storage capacity. To provide the on-line curriculum maintenance services so important to a CATSDM system, much more disc storage is needed. Fortunately the technology has advanced rapidly. For example, on today's TICCIT systems (another minicomputer-based system) over 300 million bytes of storage can be provided for about the same price as 25 million bytes only six years ago. If VTS is used, about this level additional storage (300 million bytes) is recommended. Cost if added - less than 50,000 per system.

(However, if one 300M drive is used instead of the drives currently installed with VTS, the savings might accrue.)

One other item is needed, namely advanced curriculum maintenance capabilities such as a system to allow entry of large amounts of curriculum materials on-line, the ability to search, revise, and edit this data base, and in general to manipulate it in a variety of ways. Commercially such capabilities could be provided for under $12,500/system. (Or existing VTS capabilities could be enhanced, in house, by NWC.)

Program Development Costs

The Army Research Institute is currently issuing a series of small (under $100,000) contracts intended to provide a variety of products in the area of computer aids to ISD. This program will probably spend much less than a million dollars.

The Author Management System (AMS) developed under DARPA contract was conceived, designed, developed, and implemented for under $100,000, including hardware.

Comparing the scope of these two CATSDM related efforts to the whole CATSDM system, and assuming much will be usable from these efforts at a savings over implementing them from scratch, it is estimated that a comprehensive CATSDM system, based on an enhanced VTS, can be fully implemented within NAVAIR for under two million dollars, including additional hardware.

Depending on the acceptable minimum threshold of system sophistication, the degree to which the VTS, as it is, is directly applicable to CATSDM, and depending upon the amount of easily adaptable functions from other DOD sources, the cost might be less than one million dollars. However, hardware costs will prevent this from decreasing much more, unless the standard VTS hardware mix is upgraded to take advantage of the advancing technology, in which case additional add-on hardware/system capability will not be necessary.

SECTION V

CONCLUSIONS AND RECOMMENDATIONS

This section of the report will attempt to summarize referencing previous data items as necessary) the findings of the project study group in reference to the need for CATSDM support and the current and potential provision of such support by an existing system (VTS). Some discussion will also be outlined, on a general level, of alternatives for implementing such support based on a range of reasonable assumptions, and summary recommendations will be made.

CONCLUSIONS

The analysis of CATSDM requirements and VTS proceeded systematically across a series of data items. Some conclusions are appropriate from each and will be enumerated individually. In addition, VTS analysis conclusions and the overall conclusions of the study will be given.

Data Item 0001-List of Candidate ISD Tasks

The specific tasks within the Fleet Aviation ISD (FAISD) model can be classified as to the kinds of automated support which would be appropriate for each. Seven general categories of on-line support were referenced in this classification:

On-line Support for Data Collection
On-line Support for Text Editing/Curriculum File Maintenance
On-line Support for Scheduling
On-line Support for Standardized Reports
On-line Support for Simulation/Modeling
On-line Support for Statistical Analysis
On-line Support for Data Searches and Reports

It was concluded that automated support would be helpful in every one of the 64 specific FAISD task areas. Every task was appropriate for at least four of the seven categories of potential support and the great majority would be helped in six to seven of the categories of potential support.

Data Item 0002 - Process Analysis of FAISD

The great majority of FAISD specific tasks could be sufficiently well defined in an algorithmic (flowchart) fashion, and to a level of detail as to indicate that the potential benefits from automated support would be considerable.

Data Item 0003 - Cost/Manpower Analysis of the ISD Process

(Analysis/Formatted Data Collection item only, conclusions reserved for Data Item 0004.)

Data Item 0004 - ISD Task Priorities for CATSDM

On a variety (3 factors based on 48 independent decisions for each task) of rating points, it is possible to rank the specific tasks of the FAISD model in terms of the desirability/cost benefits to be achieved from automating most or part of that task. Twenty-eight priority levels were identified ranging from a high of 49 priority points to a low of 14.

It is possible to pursue generation of CATSDM functions on a task-by-task basic (e.g., STC's Lesson Material Authoring and Formative Tryout System, programs LAP, LPP, and LTP, respectively) or on a capability-by-capability basis (e.g., Review/Comment capabilities). An overall cost benefit will be realized, however, if the high priority subsystems are developed first on hardware/software systems providing potential support in all seven basic support categories identified in Data Item 0001.

If the most cost effective subsystems are developed first, many problems will be solved which apply equally to the less cost effective subsystems, thereby effectively increasing their cost effectiveness.

Data Item 0005 - Analysis of Existing Programs for CATSDM

Most ISD support problems have been addressed in some context (often not ISD), but in isolation or in limited combinations (not as comprehensive ISD support systems).

Many of these problems have been "solved," with widely varying degrees of efficiency and effectiveness.

Many "solutions" are relatively poorly "human-engineered" in terms of usability by ISD personnel to be served. (Programs are often written for use by programmers or by experts in using the program/systems.)

No one program/system currently available addresses more than a small set of the total range of ISD support functions needed for a full and comprehensive integrated CATSDM implementation.

There are probably hundreds of tailor-made, locally developed programs/systems throughout DOD which provide scheduling support, personnel tracking, inventory, and a myriad of other support functions which could, in some way, be applicable to CATSDM.

The VTS is the best available starting point for the development of a full CATSDM system.

VTS

The VTS has the potential for forming the basis of a good CATSDM system.

Its "multi-system" approach (reliable, economical), its programming base (BASIC PLUS, easily translatable to other systems), and its relatively advanced state of development (many CATSDM functions now addressed), and its acceptance as a standard DOD training device (Navy device number 11869), make it the most viable alternative as a starting point for CATSDM development.

RECOMMENDATIONS

The final recommendations of this study will be purposely left brief and limited to the major recommendations related to what should now be done to proceed with the development of a CATSDM system.

Recommendation 1

A full CATSDM system should be developed and implemented as soon as possible.

Recommendation 2

The Versatile Training System (VTS) now being installed in NAVAIR training sites should be the development vehicle for the CATSDM system.

Recommendation 3

VTS hardware/software/operating system capabilities should be enhanced to provide all basic or additional capabilities required to develop, test, and implement a comprehensive CATSDM system.

Recommendation 4

The hardware/software development work on the CATSDM system should be carried out under the direction of NWC at China Lake in conjunction with, but not necessarily completely subordinate to or dependent upon, the ongoing NAVAIR VTS development and implementation program.

Recommendation 5

NWC staff and funding resources should be adjusted appropriately to reflect the additional scope of work. That is, the current VTS program should not suffer by an attempt to "stretch" it to include CATSDM effort within the same program parameters, without a careful examination of needs.

Recommendation 6

NWC should be supported with state of the art ISD input, CATSDM specifications, interface with current NAVAIR ISD activities, and testing and quality control services. These should be provided by an experienced ISD group with a background in, and knowledge of, developments in the field of computer applications to instructional and training problems, and a knowledge of other related DOD computer-aids-to-ISD programs.

Recommendation 7

The CATSDM development effort should take full advantage of 1) what the VTS already does, 2) what has already been developed within DOD which might apply, and 3) those functions which have CATSDM applications, currently under development within various DOD agencies.

Recommendation 8

Throughout the CATSDM development process, a continuing study should be made both of current ISD activities (especially within NAVAIR) and of other DOD initiatives related to CATSDM.

Recommendation 9

Current NAVAIR ISD efforts such as SH-2F, P-3, and E-2C should be considered as pilot/test sites for the CATSDM system, and work should begin immediately.

PROBLEM ANALYSIS PROGRAM (PAP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Problem Statement Synopsis	PSS.PA	prose description of problem in summary form.	PADB	form driven	
Program Goals Statements	PGS.PA	numbered prose list of goals of the proposed training program	PADB	form driven	will address mission and weapons system design characteristics
Sources for Program Goals	SPG.PA	numbered prose list of sources from which program goals were derived	PADB	form driven	
Program Goals Ranking	PGR.PA	prioritized list of program goals with accompanying data summary	PADB	batch entry	survey data will be batch entered and processed to yield prioritized list of goals
Existing Resources Specification	ERS.PA	description of resources already available to the program	PADB	interactive branching	user will be queried for resources in several categories: facilities, equipment, materials, personnel, services, funds
Scheduled Resources Specification	SRS.PA	description of resources that will be made available to the program.	PADB	interactive branching	user will be queried for the same categories of resources in SRS.
Unscheduled Resources Specification	URS.PA	description of resources that are not scheduled, but may be useful.	PADB	interactive branching	similar to questioning procedure in SRS and URS.

TASK LISTING PROGRAM (TLP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Task Listing Reference Numbers	TLRN.TL	multi-digit numbers separated by periods	TLDB	interactive branching	TLRN is determined using programmed decision algorithm which interacts with the user
Task Behavior Statements	TBS.TL	prose statements from 1-200 characters in length	TLDB	form driven	
Task Conditions Statements	TCS.TL	prose statements from 1-200 characters in length	TLDB	form driven	
Task Standards Statements	TSS.TL	prose statements from 1-200 characters in length	TLDB	form driven	
Task Media Selection Data	TMSD.TL	10-25 yes/no entries indicating required instructional characteristics	TLDB	form driven	

STUDENT ENTRY LEVEL PROGRAM (SELP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Terminal Objectives in previous training	TOPT.SE	prose statements of 1-200 words	SELDB	form driven	
Student Aptitude Data	SAT.SE	numeric measures of central tendancy and ranges of scores	SELDB	form driven	
Previous Job Experience	PJE.SE	prose statements of relevant job experience	SELDB	form driven	
Student Demographic Data	SDD.SE	various numeric codes as needed	SELDB	form driven	
Objectives Hierarchy Reference Numbers	OHRN.OH		SELDB	supplied from OHDB	

TASK VALIDATION AND SELECTION PROGRAM (TVSP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Task List Survey Indicator	TLSI.TV	a flag showing which tasks in TLDB will be included on the survey form	TLDB	form driven	
Task List Survey Format	TLSF.TV	a listing of the questions asked and response alternatives allowed for each task surveyed	program logic	form driven	the program will present a standard TLSF which the user will be able to verify or modify
Task Survey Demographic Variables Format	TSDVF.TV	a listing of the questions and response alternatives asked about each survey respondent	program logic	form driven	the program will present a standard TSDVF which the user will be able to verify or modify
Task List Survey Data	TLSD.TV	a listing of the responses of each respondant to the survey	TVSDB	form driven or batch entry	data may be entered by respondants directly into the system, or they can fill out an op-scannable survey form
Task Survey Data Sorting Parameters	TSDSP.TV	a specification of how the data should be summarized in relation to the demographic variables	program logic	interactive branching	through interaction with the user it will be determined what summaries of the TSVD should be produced
Task Selection Algorithm	TSA.TV	a programmed algorithm for sorting tasks into categories using the TLSD	program logic	form driven	the program will present a standard TSA which the user will be able to modify or verify
Task Selection Indicator	TSI.TV	a one digit numeric code indicating the category of training selected for a task	TLDB TVSDB	program logic	using the TSA the program will determine the type of training appropriate for each task

OBJECTIVES HIERARCHIES PROGRAM (OHP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Objectives Hierarchies Reference Numbers	OHRN.OH	multi-digit numbers separated by periods	OHDB	interactive branching	OHRN is determined using a programmed decision algorithm that interacts with the user.
Objective Hierarchies Alternate Reference Number	OHARN.OH	multi-digit numbers separated by periods	OHDB	form driven	If an OHARN appears then no other information is included in this record. If a flag appears here then it means that other objectives have this objective as the OHARN. The cross should be consulted to find out which are referenced
Objective Cross-Reference to Task Listing	OCRTL.OH	multi-digit numbers separated by periods	OHDB	interactive branching	OCRTL is determined by the program logic and is presented to the user for verification or modification
Objective Behavior Statements	OBS.OH	prose statement from 1-200 characters in length	OHDB	from driven	OBS may be entered by calling up the corresponding TBS from TLDB and modifying it
Objectives Conditions Statements	OCS.OH	prose statement from 1-200 characters in length	OHDB	form driven	OCS may be entered by calling up the corresponding TCS from TLDB and modifying it
Objective Standards Statements	OSS.OH	prose statements from 1-200 characters in length	OHDB	form driven	OSS may be entered by calling up the corresponding TSS from TLDB and modifying it
Objectives Media Selection Data	OMSD.OH	10-25 yes/no entries indicating required instructional characteristics	OHDB	form driven	OMSD may be entered by calling up the corresponding TMSD from TLDB and modifying it

EXISTING MATERIALS EVALUATION PROGRAM (EMEP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Existing Materials Reference Number	EMRN.EM	number which would indicate source of materials	EMEDB	form driven	
Existing Materials Title	EMT.EM	prose statement	EMEDB	form driven	
Existing Materials Presentation Time	EMPT.EM	number of minutes required to present materials	EMEDB	form driven	
Existing Materials Media Code	EMMC.EM	number which indicates media requirements of these materials	EMEDB	form driven	
Existing Materials Objectives	EMO.EM	prose statements	EMEDB	form driven	
Existing Materials Evaluation Data	EMED.EM	numeric data indicating effectiveness of materials	EMEDB	form driven	this would consist of a series of numeric codes that specify all critical elements of evaluation data
Objectives Hierarchy Reference Number	OHRN.OH	previously established number which links materials to program being developed	EMEDB	supplied from OHDB	

MEDIA SELECTION PROGRAM (MSP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Objectives Hierarchy Reference Number	OHRN.OH	multi-digit number separated by periods	MSDB	provided by program	the program will prompt the users input by presenting OHRN(I) for each objective
Hands on Objective Flag	HOOF.MS	single digit code; single bit (if binary choice) code	MSDB	form driven	
Required Instructional Characteristics code	RICC.MS	multi-digit code of ones (1) and zeros (0).	MSDB	form driven	
Acceptable media choices code	AMCC.MS	multi-digit code of ones and zeros.	MSDB	program determined	
Media mix Priority rules	MMPR.MS	program logic codes		interactive branching	decision rules for choosing from among acceptable media are entered for various media plans.
Hardware/Software Requirements Code	HSRC.MS	multi-digit code of ones and zeros tied to table of requirements	MSDB	form driven	

PROCUREMENT OF TRAINERS PROGRAM (PTP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Trainer Detailed Specifications	TDS.PT	list of characteristics required of acceptable trainers.	PTDB	adapted from TRDB	
Trainer Data Item Descriptions	TDIP.PT	list of reports, documents, etc. to be delivered by the contractor.	PTDB	form driven	
Trainer Contract Data Requirements	TCDR.PT	list of requirements concerning data items.	PTDB	form driven	
Trainer Proposal Evaluation Criteria	TPEC.PT	list of criteria against which technical proposals will be evaluated.	PTDB	form driven	
Trainer Detailed Proposal Requirements	TDPR.PT	outline of technical proposal format to be adhered to by bidders.	PTDB	form driven	
Trainer Proposal Evaluation Plan	TPEP.PT	list of procedures to be used in house to evaluate proposals.	PTDB	form driven	based on TPEC
Trainer Exhibits List	TEL.PT	list of documents or other data to be included for amplification or clarification	PTDB	form driven	

TRAINER REQUIREMENTS PROGRAM (TRP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Objective Hierarchy Reference Number	OHRN.TP	multi-digit and separated by periods	TRDB	supplied by other program	
Classification of Objectives	CO.TP	alpha-numeric code	TRDB	supplied by other program	data on instructional classification of objective
Hardware/Software Characteristics	HSC.TP	prose description	TRDB	interactive branching	what piece of equipment or software system is student to learn (radar system, MAD equipment, etc) and its characteristics (color, system, etc.)
Presentation Requirements	PR.TP	tables, codes, prose description	TRDB	form driven	introductory, expository, inquisitory, response detection, adaptability requirements of each device objective
Hands-On Time Requirements	HTR.TP	number indicating hours, minutes	TRDB	form driven	estimated or restriction on time allocated for objective
Proportion of Hands-On Training Time	PHT.TP	percentage	TRDB	supplied by program	estimated proportion of course time dedicated to device sessions
Characteristics of Trainer Devices	CTD.TP	tables, charts, prose	TRDB	supplied from HTDB	capability of device to meet presentation requirements
Demonstrated Effectiveness of Trainers	DET.TP	tabulated data, prose description	TRDB	supplied from HTDB	
Cost and Maintenance data	CMD.TP	charts, lists	TRDB	supplied from HTDB	data on available trainers
Constraints Data	CD.TP	tables, charts, and prose descriptions	TRDB	supplied from other programs	information on project budget, number of students, time constraints, number of sites, etc.
Selection/Comparison Algorithms	SCA.TP	coded programs	program logic	form driven	program will make comparisons and make recommendations as to mix of devices user can verify or modify
Report Format	RF.TP	formatted prose and codes	TRDB	form driven	structure and standardized report information
Trainer Device Code	TDC.TP	alpha numeric code	TRDB	interactive branching	code identifying trainer device selected for each objective
Description of Typical Training Day/Week	DTTD.TP	hours per day, days/week	TRDB	supplied from SDDB	

SYLLABUS DEVELOPMENT PROGRAM (SDP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Objectives Hierarchy Reference Number	OHRN.OH	multi-digit number separated by periods	SDDB	supplied from OHDB	automatically input and displayed as cue to user.
Acceptable Media Choices Code	AMCC.MS	multi-digit code of ones and zeroes	SDDB	supplied from MSDB	automatically input and displayed as cue to user
Critically Ranking	CR.SD	a numerical code which indicates relative criticality/difficulty of objective	SDDB	form driven	user will rank all unit objectives within course, all lesson objectives within unit, etc.
Sequencing Algorithms	SA.SD	programmed algorithms for sequencing units, lessons and segments	program logic	form driven	the program will provide a tentative sequence which the user can verify or modify as needed
Course, Unit, Lesson Number	CUL.SD	a multi-digit number separated by periods	SDDB	supplied from program	using the sequencing algorithm, the program will provide a number which the user can verify or modify as needed
Lesson Title	LT.SD	brief prose description of lesson topic	SDDB	interactive branching	
Lesson Medium (media)	LM.SD	numerical code indicating medium or media used	SDDB	interactive branching	user selects most acceptable medium/media form AMCC
Lesson Prerequisites	LP.SD	brief prose descriptions, lesson titles or CULN codes	SDDB	supplied from program	the program will provide present prerequisites in standardized form which user can modify and verify
Estimated Lesson Time	ELT.SD	estimated time in minutes	SDDB	interactive branching	
Lesson Objective	LO.SD	prose description of conditions behaviors and standards	SDDB	supplied from program	program will supply from OHDB and user can verify and modify as needed
Student Through-put Data	STD.SD	rough estimates of such things as number of students to begin course, whether overlapping scheduling to be used, etc.	SDDB	interactive branching	PAP may help user in determining student throughput data.

SYLLABUS DEVELOPMENT PROGRAM (SDP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Description of Typical Training Day/Week	DTTD.SD	number of days per week typical student spends studying; hour/ min. in a training day; facilities availability	SDDB	interactive branching	PAP may assist user in determining this variable
Placement in ordinal syllabus (time based)	POS.SD	day and hour in ordinal syllabus where lesson belongs	SDDB	supplied from program	
Segment Interrelationship code	SIC.SD	code indicating subordinate, superordinate or independent relationship of segments within lesson	SDDB	interactive branching	
Course, Unit, Lesson, Segment Number	CULS.SD	a multi-digit code separated by periods	SDDB	supplied from program	using sequencing algorithm, the program will provide a number which the user can verify or modify
Segment Title	ST.SD	brief prose description of segment topic	SDDB	interractive branching	
Segment Objective	SO.SD	prose description of conditons, behaviors and standards	SDDB	supplied from program	program will supply from OHDB and user can verify and modify as needed
Segment Medium	SM.SD	numerical code indicating medium	SDDB	interactive branching	if more than one medium used within a lesson user will need to specify medium for each segment
Calendar	C.SD	calendar indicating holidays, weekends, actual study time per day, etc.	SDDB	interactive branching	

TRAINING SUPPORT REQUIREMENTS PROGRAM (TSRP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Media Mix Designation	MMD.TS	prose description of the media that will be used and % of the course of each	TSRDB	form driven	
Objectives Per Medium	OPM.TS	number of objectives using each of the media from the mix	TSRDB	determined by program logic	the program will access AMCC data from the MSDB and assign objectives to media categories according to the MMD.
Design Manpower Requirements	DMR.TS	table of types and amounts of manpower required to design instruction.	TSRDB	determined by program logic	using standardized manpower requirements data per type of instruction and media, program will determine design manpower requirements based on the desired MMD.
Design Cost Requirements	DCR.TS	table of all costs associated with design of instruction	TSRDB	determined by program logic	manpower costs will be calculated along with costs for facilities, equipment, supplies, and other miscellaneous items yielding total cost for design activities based on desired MMD.
Development Manpower Requirements	DVMR.TS	table of types and amounts of manpower required to develop instruction	TSRDB	determined by program logic	development manpower requirements will be calculated using standardized data in a manner similar design above.
Development Cost Requirements	DVCR.TS	table of all costs associated with development of instruction	TSRDB	determined by program logic	computed in a manner similar to design costs above.
Tryout Manpower Requirements	TMR.TS	table of types and amounts of manpower required to tryout prototype instruction	TSRDB	determined by program logic	similar to design and development, with addition of students on whom the instruction will be tried.
Tryout Cost Requirements	TCR.TS	table of all costs associated with tryout of prototype instruction	TSRDB	determined by program logic	similar to design and development costs above.
Production Manpower Requirements	PMR.TS	table of types and amounts of manpower required to produce instructional materials	TSRDB	determined by program logic	calculated using the same data categories as design and development, modified to fit production personnel
Production Cost Requirements	PCR.TS	table of all costs associated with production of instructional materials	TSRDB	determined by program logic	similar to design and development above

TRAINING SUPPORT REQUIREMENTS PROGRAM (TSRP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Implementation Manpower Requirements	IMR.TS	table of types and amounts of manpower required to implement instructional program	TSRDB	determined by program logic	will calculate using data categories in tryout above, but on a scale which reflects student loading as identified in PSS and PGS from PADB
Implementation Cost Requirements	ICR.TS	table of all costs associated with implementation of instructional program	TSRDB	determined by program logic	similar to above, though costs can be projected over several years to reflect ongoing program that must consider factors such as equipment life and inflated costs over time.
Evaluation/Revision Manpower Requirements	ERM.TS	table of types and amounts of manpower required for ongoing evaluation/revision	TSRDB	determined by program logic	
Evaluation/Revision Cost Requirements	ERC.TS	table of all costs associated with evaluation/revision of program	TSRDB	determined by program logic	

LESSON SPECIFICATION PROGRAM (LSP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Course, Unit, Lesson number	CUL.SD	a multi-digit number separated by periods	LSDB	supplied from SDDB	could be automatically input and displayed as a cue or menu choice for user.
Lesson Title	LT.SD	brief prose description of lesson topic	LSDB	supplied from SDDB	could be automatically input and displayed as a cue or menu choice for user.
Lesson Objective	LO.LS	prose description of conditions behaviors and standards	LSDB	supplied from SDDB	program will supply objectives as it appears in SDDB and user can verify or modify
Lesson Introduction	LI.LS	prose description of information to be presented as introduction	LSDB	interactive branching	
Lesson Testing Logic	LTL.LS	prose description of tests to be developed	LSDB	interactive branching	user will specify data such as whether test is to mastery or diagnostic number of items, type of stems, evaluation criteria, etc.
Course, Unit, Lesson Segment number	CULS.SD	a multi-digit number separated by periods	LSDB	supplied from SDDB	could be automatically input and displayed as a cue or menu choice for user.
Segment Title	ST.SD	brief prose description of segment topic	LSDB	supplied from SDDB	
Segment Objective(s)	SO.LS	prose description of conditions behaviors and standards	LSDB	supplied from SDDB	program will supply objective as it appears in SDDB and user can verify or modify
Segment Medium	SM.SD	numerical code indicating medium	LSDB	supplied from SDDB	
Generality	G.LS	prose of variable length	LSDB	interactive branching	core information as it is to be presented to students
Generality Help	GH.LS	prose of variable length	LSDB	interactive branching	supporting information to be presented to students
Example Specification	EXS.LS	prose of variable lengths, tables	LSDB	interactive branching	description of type and number of examples
Practice/Test Specifications	PTS.LS	prose of variable length; tables	LSDB	interactive branching	description of type and number of examples of practice and test items.

LESSON SPECIFICATION PROGRAM (LSP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Special Teaching Points	STP.LS	prose of variable lengths; tables	LSDB	interactive branching	description of any special teaching points
Common Error Analysis	CEA.LS	prose of variable lengths; tables	LSDB	interactive branching	description of common student errors, related guidelines for student feedback
Graphics Specification	GRS.LS	prose of variable lengths; tables	LSDB	interactive branching	description/special instruction for creation of art, photographic, etc.
Source Reference	SR.LS	prose of variable lengths	LSDB	interactive branching	description of source materials to be used as references in developing the lesson
Objective Classification	OC.LS	alpha-numeric code	LSDB	interactive branching	classification of objective by predefined taxonomy (such as use/rule, etc.)

IMPLEMENTATION PLAN PROGRAM (IPP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Instructional System Overview	ISO.IP	brief descriptions of instructional system structure and characteristics	IPDB	interactive branching	
Resource Scheduling Data	RSD.IP	schedule of classes, with start and end dates and required resources	IPDB	interactive branching	program will ask for student throughput data and schedule facilities, equipment and personnel resources as specified in TSRDB. Attention will be called to resource inadequacies identified.
Facilities Utilization Data	FUD.IP	list of facilities and their use	IPDB	interactive branching	provides for specification of space for storage, offices, learning centers, trainer facilities, etc. IAW the TSRDB
Equipment Maintenance Utilization Data	EMU.IP	list of equipment and maintenance and utilization procedures	IPDB	interactive branching	queries user for maintenance and utilization data for equipment specified in TSRDB
Instructional Materials Management Data	IMM.IP	list of materials and provisions for replacement of consummable items.	IPDB	interactive branching	queries user for replacement and storage provisions for materials as specified SDDB.
Student Management Data	SMD.IP	provisions for student entry progress monitoring, participation in training	IPDB	interactive branching	
Records and Reports	RR.IP	list of required recordkeeping and reporting procedures	IPDB	interactive branching	program will make suggestions based on data from logic concerning service wide reporting requirements.
Staffing Management Data	SMN.IP	provisions for training, proficiency evaluation, etc. of staff	IPDB	interactive branching	

QUALITY CONTROL PROGRAM (QCP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Formative Evaluation Roles and Responsibilities	FERR.QC	tables, charts, prose descriptions, codes	QCDB	interactive branching	description of personnel involved their roles and responsibilities
Formative Evaluation Events	FEE.QC	flowcharts, prose descriptions	QCDB	form driven	descriptions of procedures to be followed, sequences of events, review points, etc.
Small Group Tryouts	SGT.QC	flowcharts, tables, prose descriptions	QCDB	interactive branching	organizing small group tryouts and prototype production; relates to LAP and LPP
Formative Evaluation Variables and Methods of Measurement	FEVM.QC	flowcharts, prose descriptions, etc.	QCDB	form driven	description of variables to be measured; of methods to use in collecting, organizing and interpreting data, forms to be used, etc.
Internal Quality Control Roles and Responsibilities	IQCR.QC	tables, charts, prose descriptions, etc.	QCDB	interactive branching	description of personnel involved, their roles and responsibilities.
Internal Quality Control Events	IQCE.QC	flowcharts, prose descriptions, etc.	QCDB	form driven	descriptions of procedures to be followed, sequences of events, management controls, revision points
Internal Quality Control Variables and Methods of Measurement	ICVM.QC	flowcharts, prose descriptions, etc.	QCDB	form driven	description of variables to be measured; of methods to use in collecting, organizing and interpreting data, forms to be used, etc.
External Quality Control Roles and Responsibilities	EQCR.QC	tables, charts, prose descriptions, etc.	QCDB	interactive branching	description of personnel involved, their roles and responsibilities
External Quality Control Events	EQCE.QC	flowcharts, prose descriptions, etc.	QCDB	form driven	description of events, procedures, management of on-going revision and updating
External Quality Control Variables and Methods of Measurement	ECVM.QC	flowcharts, prose descriptions, etc.	QCDB	form driven	description of variables to be measured; of methods to use in collecting, organizing and interpreting data, forms to be used, etc.
Quality Control Data	QCD.QC	numerical codes	QCDB	interactive branching/ batch	raw data which will be analyzed; data referenced to OHRN and CULS
Analysis of Data	AD.QC	statistical programs	QCDB	batch entry	programs which sort, do item counts and statistical analyses of data collected.

LESSON AUTHORING PROGRAM (LAP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Course, Unit, Lesson, Segment Number	CULS.SD	multi-digit number separated by periods.	LADB	supplied from LSDB	
Lesson Specification Components	*		LADB	supplied from LSDB	*this includes LT, LI, LTL, ST, LO, SO, G, GH, EXS, PTS, STP, CEA, GRS, SM, and SR
Segment Examples	SE.LA	examples to be used in segment	LADB	form driven	created by author in accordance with EXS, CEA, STP, GRS, SM, and SR
Segment Practice/ Test Items	SPT.LA	practice and test items to be used for segment, with appropriate feedback	LADB	form driven	created by author in accordance with PTS and SO
Review Comments	RC.LA	prose comments	LADB	form driven	generated by reviewing authority
Revision Specifications	RS.LT	prose statements indicating needed changes in materials	LADB	form driven and accessed from LTDB	generated by reviewing authority or other appropriate person from LTDB.
Storyboard/Script	SS.LA	formatted prose statements	LADB	form driven	created by author as required by SM and SC
Lecture outline/aids	LOA.LA	formatted prose statements	LADB	form driven	created by author as required by SM and SO
Student Worksheets	SW.LA	formatted prose statements	LADB	form driven	created by author as required by SM and SO
Tryout Procedures	TP.LA	any special procedures regarding tryout and data collection	LADB	form driven	created by author as required by SM and SO includes necessary data collection forms

LESSON PRODUCTION PROGRAM (LPP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Course, Unit, Lesson Segment Number	CULS.SD	multi-digit number separated by periods			
Production Tasks Algorithm	PTA.LP	list of tasks which can be used for scheduling/tracking production.	LPDB	interactive branching	algorithm will vary according to the medium being produced.
Production Staffing Codes	PSC.LP	numeric codes for types of production personnel required by task.	LPDB	interactive branching	personnel will be allocated for each task in PTA
Formatting Codes	FC.LP	codes which format text into appropriate form for production	LPDB	interactive branching	manipulate text to create copy that is as ready as possible for off-line production activities
Typesetting Codes	TC.LP	codes which translate copy into pasteup or camera ready masters using phototypesetter.	LPDB	interactive branching	
Technical Requirements Specifications	TRS.LP	prose statements that relate additional requirements not already specified.	LPDB	form driven	

LESSON TRYOUT PROGRAM (LTP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Course, Unit, Lesson, Segment Number	CULS.SD	multi-digit number separated by periods.	LTDB	supplied from LADB	
Tryout Schedule	TS.LT	list of dates when tryouts occurred	LTDB	form driven	
Tryout Subjects Demographic Data	TSD.LT	names and other pertinent, background data on subjects	LTDB	form driven	subjects selected in accordance with target audience characteristics in PADB and SEDB
Tryout Monitor	TM.LT	person responsible for conducting tryout and collections data.	LTDB	form driven	
Tryout Procedures	TP.LA	any special procedures regarding tryout and data collection.	LTDB	supplied from LADB	includes necessary data collection forms.
Tryout Test Data	TTD.LT	numeric test results with pass/fail indicator	LTDB	batch entry or form driven	
Tryout Performance Data	TPD.LT	results of hands on performance testing with pass/fail indicator	LTDB	batch entry or form driven	
Tryout Attitudinal Data	TAD.LT	numeric data and representative prose comments from subjects	LTDB	batch entry or form driven	
Revision Specifications	RS.LT	prose statements indicating needed changes in materials	LTDB LADB	form driven	

COMPUTER-MANAGED INSTRUCTION PROGRAM(s) (CMIP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Test Base Frame	TBF.CM	formatted prose and coded information	CMIP	interactive branching	description of test format, instructions to be given students and other consistent portions of tests
Test Item Bank	TIB.CM	formatted prose descriptions	CMIP	form driven	files of items from which specific items can be randomly selected
Student Performance Data	SPD.CM	prose descriptions	CMIP	interactive branching/ batch	student responses and other performance data
Diagnostic Data and Prescriptions	DDP.CM	prose and coded descriptions	CMIP	interactive branching/ form driven	information on interpreting student performance data, diagnosing difficulties and prescribing future activities
Student Progress	SP.CM	coded information	CMIP	supplied by program	what segments student has completed, begun, not begun; projected completion, etc.
Instructional Materials Effectiveness	IME.CM	tables, charts, prose	CMIP	supplied by program	combinations of student performance data, normative references and other such information
Resources Needed	RN.CM	alpha numeric codes	CMIP	supplied by other programs	personnel, facilities, materials, devices required for each segment/lesson

COMPUTER-ASSISTED INSTRUCTION PROGRAM (CAIP)*

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Content Files	CF.CA	formatted prose	CAIDB	supplied from LADB	
Logic and Control Files	LCF.CA	programmed logic	program logic	form driven	program defining learner control options, when learner has control, program logic when program control
Advisor	A.CA	codes and prose	CAIDB	form driven	information to student on how to use system, how well he is doing, prescriptions, deadlines, strategy templates, current status parameters, required (or desired) performance criteria.

*CAI systems will be external to CATSDM and linked to it from an interface for the most part. Separation of logic and content, potential for learner control, and on-line advice are desireable features.

HISTORICAL TRAINER DATA PROGRAM (HTDP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Characteristics of Trainer Devices	CTD.HT	tables, charts, prose	HTDB	form driven	introductory, expository, inquisitory, response detection, adaptability, etc. characteristics of systems
Demonstrated Effectiveness of Trainers	DET.HT	tabulated data, prose descriptions	HTDB	form driven	student performance data, attitudinal data, etc. collected on deviced used on other projects
Cost and Maintenance Data	CMD.HT	charts, lists	HTDB	form driven	data on costs, repairs and maintenance considerations
Comparison Algorithms	CA.HT	programmed algorithms	program logic	interactive branching	means of specifying comparisons to be made, and seeing results
Report Format	RF.HT	formatted prose and coded data	HTDB	form driven	structure and standardized report information

TACTICS PACKAGE PROGRAM (TPP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Tactics Reference Number	TRN.TP	multi-digit number separated by periods.	TPDB	form driven	
Tactic Name	TN.TP	prose statement	TPDB	form driven	
Tactic Description	TD.TP	prose statement	TPDB	form driven	
Performance Characteristics/Limits	PCL.TP	list of prose statements	TPDB	form driven	
Tactical Source Publications References	TSP.TP	list of prose statements	TPDB	form driven	
Objectives Hierarchy Reference Number(s)	OHRN.OH	multi-digit number representing related objective(s)	TPDB	interactive branching	
Course, Unit, Lesson, Segment Number(s)	CULS.SD	multi-digit number representing related course, unit, lesson, segment	TPDB	accessed by OHRN	

MASTER PLAN PROGRAM (MPP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Master Plan Introduction	MPI.MP	prose narrative describing purpose and scope, background, and objective.	MPDB	form driven	
Procurement Strategy Options	PSO.MP	prose narrative overviewing phases and procurement strategy options and rationale	MPDB	form driven	
Organization Roles and Responsibilities	ORR.MP	prose narrative defining management and support roles within the organization.	MPDB	form driven	
Detailed Phase Descriptions	DPD.MP	narrative indicating all activities in each ISD phase.	MPDB	form driven	
Program Schedules	PS.MP	tables showing overall phase and proposed activities duration.	MPDB	form driven	
Project Funding Estimate	PFE.MP	organizational, contractor, material cost items summed by phrase.	MPDB	form driven	
Project Plans	PP.MP	list of plans for reporting, use of assets, courseware distribution, management, etc.	MPDB	form driven	
Resource Commitments Specification	RCS.MP	narrative specification of organizational resources that must be committed.	MPDB	form driven	

PROCUREMENT PACKAGE PROGRAM (PPP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Detailed Specifications	DS.PP	prose listing of all required ISD activities by phase.	PPDB	adapted from DPD.MP	
Data Item Descriptions	DID.PP	list of reports, documents, etc. to be delivered by the contractor.	PPDB	form driven	
Contract Data Requirements	CDR.PP	list or requirements concerning data items.	PPDB	form driven	
Proposal Evaluation Criteria	PEC.PP	list of criteria against which technical proposals will be evaluated.	PPDB	form driven	
Detailed Proposal Requirements	DPR.PP	outline of technical proposal format to be adhered to by bidders.	PPDB	form driven	
Proposal Evaluation Plan	PEP.PP	list of procedures to be used in-house to evaluate proposals.	PPDB	form driven	based on PEC
Exhibits List	EL.PP	list of documents or other data to be included for amplification or clarification.	PPDB	form driven	

NAVY TRAINING PLAN PROGRAM (NTPP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Navy Training Plan Introduction	NTPI.NT	formatted prose narrative	NTPDB	form driven	description of background, Navy principals scope, purpose of plan;
Description of Navy-system	DS.NT	formatted prose narrative	NTPDB	supplied from other programs.	functional and operational, description of system being trained, user can verfity and modify
Contract Specific Data	CSD.NT	formatted prose narrative	NTPDB	supplied from other programs.	contract number, contractors involved, costing information, contractor services, schedules.
Operation/Maintenance/Training Manning	OMTM.NT	formatted prose narrative, charts, labels	NTPDB	form driven	Navy personnel involved in all phases, workloads, billet requirements, etc.
Trained personnel needed per year	TPPY.NT	charts	NTPDB	form driven	trained personnel needed per year to operate maintain system.
Training Requirements	TR.NT	charts	NTPDB	form driven	sites, resources, schedules, personnel needed to support training activities.
Major Decisions/ Milestones	MDM.NT	charts and prose	NTPDB	form driven	description of major milestone events and decisions, command responsible, date, etc.

PROGRESS MONITORING PROGRAM (PMP)

Data	Acronym	Description of Data	Data Base Entered	Method of Entry	Comments
Calendar Data	CD.PM	prose	PMDB	form driven	description of start date, milestones, deadlines; identification of workdays, holidays, weekends, description of working days.
Personnel	P.PM	prose and numerical codes	PMDB	form driven	personal, biographical and identification of project required skills, workload.
Master list of skills	MLS.PM	numeric codes and prose description	PMDB	interactive branching	kind of skills the project will require, each skill represented by a numerical code.
Master list of events	MLE.PM	numeric codes, prose descriptions, etc.	PMDB	form driven	a list of events, decisions, numeric codes, descriptions of event decisions, personnel involved, equipment needed, etc.
Status vector	SV. PM	codes and descriptive data	PMDB	interactive branching/ batch	whether an event in any hierarchy is begun, complete, in progress and other related data.
Event hierarchies	EH.PM	flowcharts, codes	PMDB	form driven	procedural, sequencial description of what events take place within any phase or larger event.
Equipment/Facilities	EF.PM	prose and numerical codes	PMDB	form driven	equipment, facilities needed for each event, code for each workload.
Project Accounts	PA.PM	prose and numeric description	PMDB	form driven	costing and budget considerations.

GLOSSARY

AD.QC	Analysis of Data (Quality Control Program)
AMCC.MS	Acceptable Media Choices Code (Media Selection Program)
AMS	Author Management System
C.SD	Calendar
CA.HT	Comparison Algorithms
CAI	Computer-Assisted Instruction
CAIDB	Computer-Assisted Instruction Data Base
CAIP	Computer-Assisted Instruction Program
CATSDM	Computer-Aided Training System Development and Management
CD.PM	Calendar Data
CD.TP	Contraints Data
CDR.PP	Contract Data Requirements
CEA.LS	Common Error Analysis
CMD.HT	Cost and Maintenance Data
CMIDB	Computer-Managed Instruction Data Base
CMIP	Computer-Managed Instruction Program
CODASYL	Conference on Data Systems Languages
CO.TP	Classification of Objectives
CR.SD	Criticality Ranking
CSD.NT	Contract Specific Data
CTD.HT	Characteristics of Trainer Devices
CUL.SD	Course, Unit, Lesson Number
CULS.SD	Course, Unit, Lesson, Segment Number
DCR.TS	Design Cost Requirements
DDP.CM	Diagnostic Data and Prescriptions
DET.HT	Demonstrated Effectiveness of Trainers
DID.PP	Data Item Descriptions
DMR.TS	Design Manpower Requirements
DNS.NT	Description of Navy System
DOD	Department of Defense
DPD.MP	Detailed Phase Descriptions
DS.PP	Detailed Specifications
DTTD.SD	Description of Typical Training Day/Week
DVCR.TS	Development Cost Requirements
DVMR.TS	Development Manpower Requirements
ECVM.QC	External Quality Control Variables and Methods of Measurement
EF.PM	Equipment/Facilities
EH.PM	Event Hierarchies
EL.PP	Exhibits List
ELT.SD	Estimated Lesson Time
EM	Existing Materials
EMED.EM	Existing Materials Evaluation Data
EMEDB	Existing Materials Evaluation Data Base
EMEP	Existing Materials Evaluation Program
EMMC.EM	Existing Materials Media Code
EMC.EM	Existing Materials Media Code
EMPT.EM	Existing Materials Presentation Time

EMRN.EM	Existing Materials Reference Number
EMT.EM	Existing Materials Title
EMV.IP	Equipment Maintenance/Utilization Data
EQCE.QC	External Quality Control Events
EQCR.QC	External Quality Control Roles and Responsibilities
ERC.TS	Evaluation/Revision Manpower Requirements
ERM.TS	Evaluation/Revision Manpower Requirements
ERS.PA	Existing Resources Specification
EXS.LS	Example Specifications
FAISD	Fleet Aviation Instructional Systems Development
FC.LP	Formatting Codes
FEE.QC	Formative Evaluation Events
FERR.QC	Formative Evaluation Roles and Responsibilities
FEVM.QC	Formative Evaluation Variables & Methods
FUD.IP	Facilities Utilization Data
G.LS	Generality
GH.LS	Generality Help
GRS.LS	Graphic Specifications
HOOF.MS	Hands On Objective Flag
HSC.TP	Hardware/Software Characteristics
HSRC.MS	Hardware/Software Requirements Code
HTDDB	Historical Trainer Data Data Base
HTDP	Historical Trainer Data Program
HTIC.TP	Hands-On Time Requirement
ICR.TS	Implementation Cost Requirements
ICVM.QC	Internal Quality Control Variables and Methods of Measurement
IME.CM	Instructional Materials Effectiveness
IMM.IP	Instructional Materials Management Data
IMR.TS	Implementation Manpower Requirements
IPDB	Implementation Plan Data Base
IPP	Implementation Plan Program
IQCE.QC	Internal Quality Control Events
IQCR.QC	Internal Quality Control Roles and Responsibilities
ISD	Instructional Systems Development
ISDP	Instructional Strategy Diagnostic Profile
ISO.IP	Instructional System Overview
LADB	Lesson Authoring Data Base
LAP	Lesson Authoring Program
LI.LS	Lesson Introduction
LM.SD	Lesson Medium (Media)
LO.LS	Lesson Objective
LO.SD	Lesson Objective
LOA.LA	Lecture Outline/Aids
LP.SD	Lesson Prerequisites
LPDB	Lesson Production Data Base
LPP	Lesson Production Program
LSDB	Lesson Specification Data Base
LSP	Lesson Specification Program
LTDB	Lesson Tryout Data Base
LTL.SL	Lesson Testing Logic

LTP	Lesson Tryout Program
MDM.NT	Major Decisions/Milestones
MLE.PM	Master List of Events
MLS.PM	Master List of Skills
MMD.TS	Media Mix Designator
MMPR.MS	Media Mix Priority Rules
MPDB	Master Plan Data Base
MPI.MP	Master Plan Introduction
MPP	Master Plan Program
MSDB	Media Selection Data Base
MSP	Media Selection Program
NTPDB	Navy Training Plan Data Base
NTPI.NT	Navy Training Plan Introduction
NTPP	Navy Training Plan Program
OBS.OH	Objective Behavior Statements
OC.LS	Objective Classification
OCRTL.OH	Objective Cross Reference to Task Listing
OCS.OH	Objective Conditions Statements
OHARN.OH	Objective Hierarchies Alternate Reference Number
OHDB	Objectives Hierarchy Data Base
OHP	Objectives Hierarchy Program
OHRN.OH	Objective Hierarchies Reference Number
OMSD.OH	Objective Media Selection Data
OMTM.NT	Operation/Maintenance/Training Manning
OPM.TS	Objective Per Media
ORR.MP	Organization Roles and Responsibilities
OSS.OH	Objective Standards Statements
P.PM	Personnel
PA.PM	Project Accounts
PADB	Problem Analysis Data Base
PAP	Problem Analysis Program
PCL.TP	Performance Characteristics/Limits
PCR.TS	Production Cost Requirements
PEP.PP	Proposal Evaluation Criteria
PFE.MP	Project Funding Estimate
PGR.PA	Program Goals Ranking
PGS.PA	Program Goals Statements
PHT.TP	Proportion of Hands-On Training Time
PJE.SE	Previous Job Experience
PMDB	Progress Monitoring Data Base
PMP	Progress Monitoring Program
PMR.TS	Production Manpower Requirements
POS.SD	Placement in Ordinal Syllabus (Time-based)
PP.MP	Project Plans
PPDB	Procurement Package Data Base
PPP	Procurement Package Program
PQS	Personnel Qualification System
PR.TP	Presentation Requirements
PS.MP	Program Schedules
PSC.LP	Production Staffing Codes
PSO.MP	Procurement Strategy Options

PSS.PA	Problem Statement Synopsis
PTA.LP	Production Tasks Algorithms
PTDB	Procurement of Trainers Data Base
PTP	Procurement of Trainers Program
PTS.LS	Practice/Test Specifications
QCDB	Quality Control Data Base
QCP	Quality Control Program
RC.LA	Review Comments
RCS.MP	Resource Commitments Specification
RF.HT	Report Format
RICC.MS	Required Instructional Characteristics Code
RN.CM	Resources Needed
RR.IP	Records and Reports
RS.LA	Revision Specifications
RSD.IP	Resource Scheduling Data
SA.SD	Sequencing Algorithms
SAT.SE	Student Aptitude Data
SCA.TP	Selection/Comparison Algorithms
SDD.SE	Student Demographic Data
SDDB	Syllabus Development Data Base
SDP	Syllabus Development Program
SE.LA	Segment Examples
SELDB	Student Entry-Level Data Base
SELP	Student Entry-Level Program
SGT.QC	Small Group Tryouts
SIC.SD	Segment Interrelationship Code
SM.SD	Segment Medium
SMD.IP	Student Management Data
SMN.IP	Staffing Management Data
SO.LS	Segment Objective
SO.SD	Segment Objective
SP.CM	Student Progress
SPD.CM	Student Performance Data
SPG.PA	Sources for Program Goals
SPT.LA	Segment Practice/Test Items
SR.LS	Source References
SRS.PA	Scheduled Resources Specification
SS.LA	Storyboard/Script
ST.SD	Segment Title
STD.SD	Student Through-put Data
STP.LS	Special Teaching Points
SV.PM	Status Vector
SW.LA	Student Worksheets
TAD.LT	Tryout Attitudinal Data
TBF.CM	Test Base Frame
TBS.TL	Task Behavior Statements
TC.LP	Typesetting Codes
TCDR.PT	Trainer Contract Data Requirements
TCR.TS	Tryout Cost Requirements
TCS.TL	Task Conditions Statements
TD.TP	Tactic Description

TDC.TP	Trainer Device Code
TDID.PT	Trainer Data Item Descriptions
TDPR.PT	Trainer Detailed Proposal Requirements
TDS.PT	Trainer Detailed Specifications
TEL.PT	Trainer Exhibits List
TIB.CM	Test Item Bank
TLDB	Task Listing Data Base
TLP	Task Listing Program
TLRN.TL	Task Listing Reference Number
TLSD.TV	Task List Survey Data
TLSF.TV	Task List Survey Format
TLSI.TV	Task List Survey Indicator
TM.LT	Tryout Monitor
TMR.TS	Tryout Manpower Requirements
TMSD.TL	Task Media Selection Data
TN.TP	Tactic Name
TOPT.SE	Terminal Objectives From Previous Training
TP.LA	Tryout Procedures
TPD.LT	Tryout Performance Data
TPDB	Tactics Package Data Base
TPEC.PF	Trainer Proposal Evaluation Criteria
TPEP.PT	Trainer Proposal Evaluation Plan
TPP	Tactics Package Program
TPPD	Trainer Procurement Package Development
TPPY.NT	Trained Personnel Needed Per Year
TR.NT	Training Requirements
TRD	Trainer Requirements Determination
TRDB	Trainer Requirements Data Base
TRN.TP	Tactics Reference Number
TRP	Trainer Requirements Program
TRS.LP	Technical Requirements Specifications
TS.LT	Tryout Schedule
TSA.TV	Task Selection Algorithm
TSD.LT	Tryout Subjects Demographic Data
TSDSP.TV	Task Survey Data Summarizing Parameters
TSDVF.TV	Task Survey Demographic Variables Format
TSI.TV	Task Selection Indicator
TSP.TP	Tactical Source Publications References
TSRDB	Training Support Requirements Data Base
TSRP	Training Support Requirements Program
TSS.TL	Task Standards Statements
TTD.LT	Tryout Test Data
TVSDB	Task Validation and Selection Data Base
TVSP	Task Validation and Selection Program
URS.PA	Unscheduled Resources Specification
VTS	Versatile Training System